

04/07/00
jc777 U.S. PTO

04-10-00

A

Case Docket No.: 7165

Patent Department
ZENITH ELECTRONICS CORPORATION
1000 Milwaukee Avenue
Glenview, Illinois 60025-2493
(847) 391-8014

Box PATENT APPLICATION
Assistant Commissioner for Patents
Washington, D.C. 20231

jc564 U.S. PTO
09/545597
04/07/00

Sir:

Transmitted herewith for filing is the patent application of

Inventors: Richard W. Citta
Scott M. LoPresto
Jingsong Xia

For: MULTIPATH GHOST ELIMINATING EQUALIZER WITH OPTIMUM
NOISE ENHANCEMENT

Enclosed are specification, claims, Declaration and Power of Attorney and 17
pages of informal drawings.

FEE CALCULATION

CLAIMS AS FILED					
	FOR	NUMBER FILED	NUMBER EXTRA	RATE	BASIC FEE \$690.00
	TOTAL CLAIMS (20 allowed)	93	73	\$18.00	\$1,314.00
	INDEPENDENT CLAIMS (3 allowed)	3	0	\$78.00	0
				TOTAL FILING FEE	\$2,004.00

Please charge Deposit Account No. 26-0175 in the amount of \$2,004.00. The
Commissioner is hereby authorized to charge any additional fees which may be
required, or credit any overpayment to this account. A duplicate copy of this
sheet is enclosed.

References attached.

Jack Nail

Attorney of Record
Reg. No. 27,566

Express Mail mailing label number: EM588929304US

Date of Deposit: *April 7, 2000*

I hereby certify that this paper or fee is being deposited with the United States Postal Service "Express Mail Post Office to Addressee"
service under 37 CFR 1.10 on the date indicated above and is addressed to the Assistant Commissioner for Patents, Box PATENT
APPLICATION, Washington, D.C. 20231

Patricia A. Wiederer

(Type or printed Name of Person Mailing paper or fee)

Patricia A. Wiederer

(Signature of person mailing paper or fee)

MULTIPATH GHOST ELIMINATING EQUALIZER
WITH OPTIMUM NOISE ENHANCEMENT

Technical Field of the Invention

5 The present invention is directed to an equalizer that substantially eliminates signal ghosts of up to and including 100% ghosts and, more particularly, to a multipath equalizer with optimum noise enhancement.

Background of the Invention

10 Ghosts are produced at a receiver usually because a signal arrives at the receiver through different transmission paths. For example, in a system having a single transmitter, the multipath transmission of a signal may occur because of signal reflection. That is, the receiver receives a transmitted signal and one or more reflections (i.e., ghosts) of the transmitted signal. As
15 another example, the multipath transmission of a signal may occur in a system having plural transmitters that transmit the same signal to a receiver using the same carrier frequency. A network which supports this type of
20 transmission is typically referred to as a single frequency network. Because a ghost results from the multipath

transmission of a signal, a ghost is often referred to as multipath interference.

5 A variety of systems have been devised to deal with the problems caused by ghosts. For example, spread spectrum systems deal very adequately with the problem of a 100% ghost by spreading the transmitted data over a substantial bandwidth. Accordingly, even though a 100% ghost means that some information may be lost, the data can still be recovered because of the high probability that it was spread over frequencies that were not lost because of the ghost. Unfortunately, the data rate associated with spread spectrum systems is typically too low for many applications.

10 It is also known to transmit data as a data vector. A matched filter in a receiver correlates the received data with reference vectors corresponding to the possible data vectors that can be transmitted. A match indicates the transmitted data. Unfortunately, the data rate typically associated with the use of matched filters is still too low for many applications.

20 When high data rates are required, equalizers are often used in a receiver in order to reduce ghosts of a main

signal. An example of a time domain equalizer is a FIR filter. A FIR filter convolves its response with a received signal in order to recover data and eliminate any ghosts of the data. The coefficients applied by the FIR filter asymptotically decrease toward zero for ghosts that are less than 100%. However, for 100% ghosts, the coefficients applied by the FIR filter do not asymptotically decrease toward zero so that a FIR filter becomes infinitely long if a 100% ghost is to be eliminated, making the FIR filter impractical to eliminate a 100% ghost.

A frequency domain equalizer typically includes a Fast Fourier Transform (FFT) which is applied to the received signal. A multiplier multiplies the frequency domain output of the FFT by a compensation vector. An inverse FFT is applied to the multiplication results in order to transform the multiplication results to the time domain. The compensation vector is arranged to cancel the ghost in the received signal leaving only the main signal. However, information in the received main signal is lost at certain frequencies so that the output of the inverse FFT becomes only an approximation of the transmitted data.

U.S. Application 09/158,730 filed September 22, 1998 discloses a vector domain equalizer which effectively eliminates ghosts up to 100% by distributing the transmitted data in both time and frequency so that the vectors are essentially random in the time and frequency domains. Accordingly, in a heavily ghosted channel, all data can be recovered with small noise enhancement, and any enhanced noise that does exist is near white. However, the number of calculations performed by the transform in the receiver to recover the data is large.

U.S. Application Serial No. 09/283,877 filed April 1, 1999 discloses a single path equalizer which effectively eliminates ghosts up to 100% and which uses fewer calculations. This equalizer includes a pre-processor, a finite filter, and a post-processor. The pre-processor multiplies a data input block received from the channel by coefficients **b** in order to modulate the received main signal and its ghost so that the ghost is less than the received main signal. The finite filter applies coefficients **a** in order to eliminate the ghost from the multiplication results. The post-processor applies coefficients **c** to the output of the finite filter in order to reverse the effects

of the modulation imposed by the pre-processor. Also, the post-processor applies a window function to the output of the finite filter. This single path equalizer somewhat enhances noise picked up in the channel.

5 U.S. Application Serial No. 09/425,522 filed October 22, 1999 discloses a dual parallel path equalizer having a pre-processor, a finite filter, and a post-processor in each path. The pre-processors multiply a data input block received from the channel by corresponding coefficients b_1 and b_2 in order to modulate the received main signal and its ghost so that the ghost is less than the received main signal. The finite filters apply corresponding coefficients a_1 and a_2 in order to eliminate the ghost from the multiplication results. The post-processors apply corresponding coefficients c_1 and c_2 to the outputs of the finite filters in order to reverse the effects of the modulations imposed by the pre-processors. Each of the outputs of the post-processors is a solution to the problem of a ghost. That is, substantially no ghost is present in the output from each of the post-processors. The outputs of the post-processors are added in order to substantially minimize enhancement of noise, thus producing

better signal to noise performance as compared to a single path equalizer.

The present invention is directed to an equalizer and to a controller for controlling the equalizer so that the equalizer has less hardware than past equalizers and so that the equalizer converges onto an output in which ghosts up to 100% are substantially eliminated.

Summary of the Invention

In accordance with one aspect of the present invention, a method is provided to equalize a signal. The signal comprises a series of input blocks of coded data. The method comprises at least the following steps: a) shifting data in each input block of data left by one; b) complex multiplying each of the left shifted input block of data by a first set of equalizer coefficients to provide respective first adjusted output blocks of data, wherein step b) is not a full solution to ghosts; c) complex multiplying each of the input blocks of data by a second set of equalizer coefficients to provide respective second adjusted output blocks of data, wherein step c) is not a full solution to ghosts; d) shifting the data in each input

block of data right by one; e) complex multiplying each of the right shifted input block of data by a third set of equalizer coefficients to provide respective third adjusted output blocks of data, wherein step e) is not a full
5 solution to ghosts; f) adding corresponding ones of the first, second, and third adjusted output blocks of data; and, g) controlling the first, second, and third sets of equalizer coefficients so that, as a result of the addition performed according to step f), a substantially full
10 solution to ghosts is obtained.

In accordance with a further aspect of the present invention, an equalizer for processing blocks of data comprises $n - 1$ data shifters, n finite filters, an adder, and a controller. Each of the $n - 1$ data shifters shifts the blocks of data. One of the n finite filters applies a corresponding set of finite filter coefficients to the blocks of data, and each of the other $n - 1$ finite filters applies a set of finite filter coefficients to a
15 corresponding output of the $n - 1$ data shifters. Ghosts of the blocks of data are not eliminated as a result of the application of the sets of finite filter coefficients corresponding to the n finite filters, and $n > 2$. The adder
20

is arranged to add outputs from the n finite filters. The controller is arranged to control the sets of finite filter coefficients corresponding to the n finite filters so that the addition performed by the adder substantially eliminates the ghosts.

In accordance with another aspect of the present invention, an equalizer for processing blocks of data comprises first, second, third, and fourth data shifters, first, second, third, fourth, and fifth finite filters, an adder, and a controller. The first data shifter is arranged to shift the data left by two, and the second data shifter is arranged to shift the data left by one. The third data shifter is arranged to shift the data right by one, and the fourth data shifter is arranged to shift the data right by two. The first finite filter applies a first set of finite filter coefficients to each of the blocks of data which have been shifted by the first data shifter. Ghosts of the blocks of data are not eliminated as a result of the application of the first set of finite filter coefficients. The second finite filter applies a second set of finite filter coefficients to each of the blocks of data which have been shifted by the second data shifter. Ghosts of the

blocks of data are not eliminated as a result of the application of the second set of finite filter coefficients. The third finite filter applies a third set of finite filter coefficients to each of the blocks of data. Ghosts of the blocks of data are not eliminated as a result of the application of the third set of finite filter coefficients. The fourth finite filter applies a fourth set of finite filter coefficients to each of the blocks of data which have been shifted by the third data shifter. Ghosts of the blocks of data are not eliminated as a result of the application of the fourth set of finite filter coefficients. The fifth finite filter applies a fifth set of finite filter coefficients to each of the blocks of data which have been shifted by the fourth data shifter. Ghosts of the blocks of data are not eliminated as a result of the application of the fifth set of finite filter coefficients. The adder is arranged to add outputs from the first, second, third, fourth, and fifth finite filters. The controller is arranged to control the first, second, third, fourth, and fifth sets of finite filter coefficients so that the addition performed by the adder substantially eliminates the ghosts.

Brief Description of the Drawings

These and other features and advantages of the present invention will become more apparent from a detailed consideration of the invention when taken in conjunction with the drawings in which:

Figure 1 illustrates a first embodiment of an equalizer synchronization and coefficient control according to the present invention;

Figure 2 illustrates a second embodiment of an equalizer synchronization and coefficient control according to the present invention;

Figure 3 illustrates a first embodiment of an equalizer in the form of a dual path equalizer according to the present invention;

Figure 4 illustrates an exemplary set of responses for corresponding elements of Figure 3;

Figure 5 illustrates a second embodiment of an equalizer in the form a dual path equalizer according to the present invention;

Figure 6 illustrates an exemplary set of responses for corresponding elements of Figure 5;

Figure 7 illustrates a third embodiment of an equalizer in the form a dual path equalizer according to the present invention;

5 Figure 8 illustrates a fourth embodiment of an equalizer in the form a triple path equalizer according to the present invention;

Figure 9 illustrates a fifth embodiment of an equalizer in the form a triple path equalizer according to the present invention;

10 Figure 10 illustrates a flow graph for the triple path equalizer shown in Figure 9;

Figure 11 illustrates an exemplary set of responses for corresponding elements of Figure 9;

15 Figure 12 illustrates a sixth embodiment of an equalizer in the form a generalized multipath equalizer according to the present invention;

Figure 13 illustrates a seventh embodiment of an equalizer in the form a generalized multipath equalizer according to the present invention;

20 Figure 14 shows a first embodiment of a converger arranged to control convergence of the dual path equalizer shown in Figure 3;

Figure 15 shows a second embodiment of a converger arranged to control convergence of the dual path equalizer shown in Figure 5;

Figure 16 shows a fourth embodiment of a converger arranged to control convergence of the triple path equalizer shown in Figure 8;

Figure 17 shows a fifth embodiment of a converger arranged to control convergence of the triple path equalizer shown in Figure 9; and,

Figure 18 shows an exemplary error generator that may be used for the error generators of the convergence controls shown in Figures 14, 15, 16, and 17.

Detailed Description

SYSTEM DIAGRAMS

As discussed below, the equalizers of the present inventions apply **b**, **a**, and **c** coefficients to received data blocks in order to substantially eliminate ghosts of the data blocks that may be received by a receiver. The **c** coefficients are fixed in both magnitude and width and, thus, require only synchronization with the data blocks. The **b** coefficients are varied in width as the time interval

between data blocks and ghosts vary. Thus, the **b** coefficients require both synchronization and width control. The **a** coefficients are varied in width, magnitude, and phase according to both the time interval between data blocks and ghosts and according to the characteristics of the ghosts. Accordingly, the **a** coefficients require synchronization, and width, magnitude, and phase control.

A first embodiment of a synchronization and coefficient control is shown in Figure 1. As shown in Figure 1, an equalizer 1, such as any of the equalizers described below, receives data and ghosts of the data. The data, for example, may be transmitted in the form of data blocks. The equalizer 1 processes the data blocks and ghosts in order to eliminate ghosts. A synchronizer 2 synchronizes the equalizer 1 to the incoming data blocks using any conventional synchronizing technique. A channel estimator 3 estimates the characteristics of the channel through which the data blocks are transmitted in order to control the **b** coefficients in width and to control the **a** coefficients in width, magnitude, and phase so as to substantially eliminate ghosts.

09/04/2009 10:59:59 AM

The channel estimator 3, for example, may be any of the channel estimators that are conventionally used in COFDM systems. Such channel estimators may be used to estimate the magnitude of ghosts, and the time interval between data blocks and their ghosts. The magnitude and time interval can be used as addresses into look up tables in order to read out the appropriate sets of **a** coefficients. In order to reduce the size of such look up tables to a manageable number of entries, interpolation may be used between entries when a combination of the time interval **d** and ghost magnitude does not correspond exactly to the addresses of the look up tables. The channel estimator 3 also uses the time interval **d** to control the width of the **b** coefficients as described below.

15
20

A second embodiment of a synchronization and coefficient control is shown in Figure 2. As shown in Figure 2, an equalizer 5, such as any of the equalizers described below, receives data and ghosts. The data, for example, may be in the form of data blocks. The equalizer 5 processes the data blocks and ghosts in order to eliminate the ghosts. A synchronizer 6 synchronizes the equalizer 5 to the incoming data blocks using any conventional

synchronizing technique. A converger 7 compares the input and output of the equalizer 5 and adjusts the **a** coefficients based upon the comparison results in a direction to substantially eliminate ghosts. The converger 7 also determines the time interval d between data blocks and ghosts in order to control the width of the **b** coefficients. The converger 7 is described in more detail below.

EQUALIZERS

A first equalizer embodiment, i.e., a dual path equalizer 10, is shown in Figure 3 and includes a pre-processor 12 and a 2xFFT 14 (i.e., a twice-the-data-block-length FFT 14) in a common leg of the dual path equalizer 10. The dual path equalizer 10 also includes a first finite filter 16, a first $2xFFT^{-1}$ 18, and a first post-processor 20 in a first path 22, and a second finite filter 24, a second $2xFFT^{-1}$ 26, and a second post-processor 28 in a second path 30. The outputs of the first and second post-processors 20 and 28 are added by an adder 32.

The pre-processor 12 of the dual path equalizer 10 multiplies the signal received from the channel by coefficients b_0 . The data may be transmitted in blocks with

09545597.040700
10
15
a guard interval between each adjacent pair of data blocks. The signal processed by the dual path equalizer 10, therefore, includes the data blocks and any ghosts of the data blocks. The coefficients b_0 are arranged as a window function that is substantially coextensive with a received data block and its ghost. Thus, the pre-processor 12 eliminates any energy, primarily noise, that is outside of the data blocks and their ghosts. The signal received from the channel is designated in Figure 3 as Input Data.

The 2xFFT 14 applies a Fast Fourier Transform to the output of the pre-processor 12. The Fast Fourier Transform has sufficient coefficients so that it is longer than a data block and may be up to twice as long as a data block.

The first finite filter 16 applies coefficients A_1 to the output of the 2xFFT 14. The first finite filter 16 may be implemented as a complex multiplier that complex multiplies the coefficients A_1 by the frequency domain output of the 2xFFT 14. (Upper case letters are used to denote the frequency domain, and lower case letters are used to denote the time domain.) The output of the first finite filter 16 includes the data, a modified ghost of the data,

and enhanced noise. Unlike the finite filters of the dual path equalizer disclosed in the aforementioned U.S.

Application Serial No. 09/425,522, the first finite filter 16 does not eliminate the ghost from the frequency domain output of the pre-processor 12.

The first $2x\text{FFT}^{-1}$ 18 applies an Inverse Fast Fourier Transform to the output of the first finite filter 16. The Inverse Fast Fourier Transform has sufficient coefficients so that it is longer than a data block and may be up to twice as long as a data block.

The first post-processor 20 multiplies the time domain output from the first $2x\text{FFT}^{-1}$ 18 by coefficients \mathbf{c}_1 . The first post-processor 20 performs a window function to eliminate any energy, primarily noise, outside of the data blocks. This window function has a duration (i.e., width) which is substantially equal to the duration (i.e., width) of a data block. Also, the coefficients \mathbf{c}_1 applied by the first post-processor 20 are chosen so as to substantially minimize noise enhancement. Unlike the output of the first post-processor of the dual path equalizer disclosed in the aforementioned U.S. Application Serial No. 09/425,522, the

output of the first post-processor 20 does not, by itself, represent a solution to the problem of a ghost.

5 The second finite filter 24 applies coefficients A_2 to the output of the 2xFFT 14. The output of the second finite filter 24 includes the data of the data block, a modified ghost of the data block, and noise. As in the case of the first finite filter 16, the second finite filter 24 may be implemented as a complex multiplier and does not eliminate the ghost from the frequency domain output of the 2xFFT 14.

10 The second $2xFFT^{-1}$ 26 applies an Inverse Fast Fourier Transform to the output of the second finite filter 24. The Inverse Fast Fourier Transform has sufficient coefficients so that it is longer than a data block and may be up to twice as long as a data block.

15 The second post-processor 28 multiplies the time domain output from the second $2xFFT^{-1}$ 26 by coefficients c_2 . The second post-processor 28 performs a window function to eliminate any energy in the received signal that is outside of the data blocks. This window function has a duration which is substantially equal to the duration of a data block. Also, the coefficients c_2 applied by the second

20

post-processor 28 are chosen so as to substantially minimize noise enhancement. As in the case of the first post-processor 20, the output of the second post-processor 28 does not represent a solution to the problem of a ghost.

5 As discussed above, the outputs of the first and second post-processors 20 and 28 are added by the adder 32. As a result, the data blocks emerging from each of the first and second paths 22 and 30 are correlated and add to produce a larger amplitude. The ghosts in the outputs of the first and second post-processors 20 and 28 substantially cancel because of the application of the coefficients A_1 , A_2 , c_1 , and c_2 . Noise (such as white noise) is less correlated than the data blocks so that, when the outputs of the first and second post-processors 20 and 28 are added, the noise adds to a less extent than does the data blocks. Thus, noise enhancement is substantially minimized.

10
15
20 Thus, although neither of the outputs of the first and second post-processors 20 and 28 is a solution to channel distortion such as ghosts up to 100%, the combined output from the adder 32 is a solution, so that the dual path equalizer 10 adequately deals with ghosts up to and including a 100% ghost. Additionally, the signal to noise

ratio of the dual path equalizer 10 is improved over a single path equalizer.

Exemplary sets of the coefficients b_0 , A_1 , A_2 , c_1 , and c_2 are shown in Figure 4. The first column of Figure 4 shows the real parts of the coefficients b_0 , A_1 , A_2 , c_1 , and c_2 , the second column of Figure 4 shows the imaginary parts of the coefficients b_0 , A_1 , A_2 , c_1 , and c_2 , and the third column of Figure 4 shows the absolute values of the coefficients b_0 , A_1 , A_2 , c_1 , and c_2 .

As can be seen from Figure 4, and as discussed above, the coefficients b_0 applied by the pre-processor 12 have a unity real part and a zero imaginary part that are arranged as a window function having a duration that is substantially coextensive with a received data block and its ghost. The number of coefficients in the set of coefficients b_0 depends upon the size of the data block and the interval d between the signal and its ghost. As shown in Figure 4, for example, if there are sixteen samples in a data block and the interval d between the data block and its ghost is 1 sample (i.e., $1/16$ of a data block), then there are seventeen coefficients ($16 + 1$) in the set of coefficients b_0 . The width of (i.e., the number of

coefficients in) the set of coefficients b_0 , therefore, is varied as the interval d between the data block and its ghost varies.

5 The coefficients c_1 and c_2 applied respectively by the first and second post-processors 20 and 28 have oppositely sloped real parts and zero imaginary parts that are arranged as corresponding window functions. The coefficients c_1 and c_2 are sloped, linear, and reversed weighting functions that operate to optimize noise in such a way that the signal to noise ratio at the output of the adder 32 is substantially enhanced. The width of the coefficients c_1 is fixed and is equal to the width of a data block. Thus, if a data block has sixteen samples, then there are sixteen coefficients c_1 . Similarly, the width of the coefficients c_2 is fixed and is equal to the width of a data block. Thus, if a data block has sixteen samples, then there are sixteen coefficients c_2 .

15
20 The coefficients A_1 and A_2 applied respectively by the first and second finite filters 16 and 24 have non-zero real and imaginary parts. As discussed below, the coefficients A_1 and A_2 are adjusted during operation of the dual path equalizer 10 so that, when the outputs of the

first and second paths 22 and 30 are added by the adder 32, ghosts are substantially eliminated and noise enhancement is substantially minimized. Each of the coefficients A_1 and A_2 has a width that is up to twice as long as the width of a data block. Thus, if the data block has sixteen samples, for example, there may be up to thirty-two coefficients for each of the coefficients A_1 and A_2 .

The pre-processor 12 eliminates noise that may exist outside of the data block and its ghost. Thus, the pre-processor 12 could be eliminated from the dual path equalizer 10. However, if so, noise in the output of the dual path equalizer 10 will be greater than if the pre-processor 12 were not eliminated.

The dual path equalizer 10 includes only one pre-processor and only one FFT. Accordingly, there is less hardware in the dual path equalizer 10 than in the dual path equalizer disclosed in the aforementioned application 09/425,522.

As a second equalizer embodiment, a dual path equalizer 40 is shown in Figure 5 and includes a pre-processor 42 and a 2xFFT 44 in a common leg. The dual path equalizer 40 also includes a first finite filter 46 and a

first post-processor 48 in a first path 50, and a second
finite filter 52 and a second post-processor 54 in a second
path 56. The outputs of the first and second post-
processors 48 and 54 are added by an adder 58, and the
5 output of the adder 58 is down sampled by two by a down
sampler 59. Down sampling is needed to reduce the over
sampling data produced by the 2xFFT 44 in a manner
equivalent to that effected by the window functions applied
by the first and second post-processors 20 and 28 of Figure
3.

The pre-processor 42 of the dual path equalizer 40
multiplies the signal received from the channel by
coefficients b_0 . As in the case of the coefficients b_0
applied by the pre-processor 12, the coefficients b_0 applied
by the pre-processor 42 are arranged as a window function
15 that is substantially coextensive with a received data block
and its ghost.

The 2xFFT 44 applies a Fast Fourier Transform to
the output of the pre-processor 42. The Fast Fourier
20 Transform has sufficient coefficients so that it is longer
than a data block and may be up to twice as long as a data
block.

The first finite filter 46 applies coefficients A_1 to the output of the 2xFFT 44. The first finite filter 46 may be implemented as a complex multiplier that complex multiplies the coefficients A_1 by the frequency domain output of the 2xFFT 44. As in the case of the first and second finite filters 16 and 24 of Figure 3, the first finite filter 46 does not eliminate ghosts from the frequency domain output of the 2xFFT 44.

The first post-processor 48 convolves the frequency domain output from the first finite filter 46 with coefficients C_1 . As in the case of the first and second post-processors 20 and 28 of Figure 3, the output of the first post-processor 48 does not represent a solution to the problem of a ghost. However, the first post-processor 48 weights the output of the first finite filter 46 in order to substantially optimize noise in the first path 50. Accordingly, when the output of the first post-processor 48 is combined with the output of the second post-processor 54 by the adder 58, ghosts are substantially eliminated in the output of the adder 58, and noise enhancement is substantially minimized.

5 The second finite filter 52 applies coefficients A_2 to the output of the 2xFFT 44. The second finite filter 52 may be implemented as a complex multiplier that complex multiplies the coefficients A_2 by the frequency domain output of the 2xFFT 44. As in the case of the first finite filter 46, the second finite filter 52 does not eliminate ghosts from the frequency domain output of the pre-processor 42.

10 The second post-processor 54 convolves the frequency domain output from the second finite filter 52 with coefficients C_2 . As in the case of the first post-processor 48, the output of the second post-processor 54 does not represent a solution to the problem of a ghost. However, the second post-processor 54 weights the output of the second finite filter 52 in order to substantially optimize the noise picked up from the channel by the received signal. Accordingly, as discussed above, when the output of the first post-processor 48 is combined with the output of the second post-processor 54 by the adder 58, ghosts are substantially eliminated in the output of the adder 58, and noise enhancement is substantially minimized.

As indicated above, the outputs of the first and second post-processors 48 and 54 are added by the adder 58. As a result, the data blocks emerging from each of the first and second paths 50 and 56 are correlated and add to produce a larger amplitude. The ghosts in the outputs of the first and second post-processors 48 and 54 substantially cancel because of the application of the coefficients A_1 , A_2 , C_1 , and C_2 . Noise (such as white noise) is less correlated than the data blocks so that, when the outputs of the first and second post-processors 48 and 54 are added, the noise adds to a less extent than does the data blocks. Thus, noise enhancement is substantially minimized.

Accordingly, although neither of the outputs of the first and second post-processors 48 and 54 is a solution to channel distortion such as ghosts up to the 100%, the combined output from the adder 58 is a solution, so that the dual path equalizer 40 adequately deals with ghosts up to and including a 100% ghost. Additionally, the signal to noise ratio of the dual path equalizer 40 is improved over a single path equalizer.

The dual path equalizer 40 does not include the first $2 \times \text{FFT}^{-1}$ 18 and the second $2 \times \text{FFT}^{-1}$ 26 of the dual path

equalizer 10. Accordingly, there is less hardware in the dual path equalizer 40 than in the dual path equalizer 10. Also, because there is a Fast Fourier Transform but no Inverse Fast Fourier Transform upstream of the first and second post-processors 48 and 54, the first and second post-processors 48 and 54 are arranged to operate in the frequency domain. Because the dual path equalizer 40 has an FFT but no inverse FFTs, an inverse FFT should be included in the transmitter that transmits the data blocks to the dual path equalizer 40.

Exemplary sets of the coefficients b_0 , A_1 , A_2 , C_1 , and C_2 are shown in Figure 6 for an interval or delay of $1/16$ between a data block and its ghost. The first column of Figure 6 shows the real parts of the coefficients b_0 , A_1 , A_2 , C_1 , and C_2 , the second column of Figure 6 shows the imaginary parts of the coefficients b_0 , A_1 , A_2 , C_1 , and C_2 , and the third column of Figure 6 shows the absolute values of the coefficients b_0 , A_1 , A_2 , C_1 , and C_2 . The coefficients b_0 applied by the pre-processor 42 may be the same as the coefficients b_0 applied by the pre-processor 12. As can be seen from Figure 6, and as discussed above, the coefficients b_0 applied by the pre-processor 42 have a unity real part

and a zero imaginary part that are arranged as a window function having a duration that is substantially coextensive with a received data block and its ghost.

5 The coefficients A_1 and A_2 applied respectively by the first and second finite filters 46 and 52 have non-zero real and imaginary parts. As discussed below, the coefficients A_1 and A_2 are adjusted during operation of the dual path equalizer 40 so that, when the outputs of the first and second paths 50 and 56 are added by the adder 58, ghosts are substantially eliminated and noise enhancement is substantially minimized.

10 The coefficients C_1 applied by the first post-processor 48 may be derived by performing a Fast Fourier Transform of the coefficients c_1 applied by the first post-processor 20 of the dual path equalizer 10. A predetermined number of the resulting coefficients may then be used as the coefficients C_1 . For example, as shown in Figure 6, the resulting seven most significant coefficients are used as the coefficients C_1 . However, more of the coefficients
15
20 resulting from the Fast Fourier Transform of the coefficients c_1 applied by the first post-processor 20 could

be used in order to more effectively eliminate ghosts and minimize noise enhancement.

Similarly, the coefficients C_2 applied by the second post-processor 56 may be derived by performing a Fast Fourier Transform of the coefficients c_2 applied by the second post-processor 28 of the dual path equalizer 10. A predetermined number of the resulting coefficients may then be used as the coefficients C_2 . For example, as shown in Figure 6, the resulting seven most significant coefficients are used as the coefficients C_2 . However, more of the coefficients resulting from the Fast Fourier Transform of the coefficients c_2 applied by the second post-processor 28 could be used if desired.

As a third equalizer embodiment, a dual path equalizer 60 is shown in Figure 7 and includes a pre-processor 62 and a 2xFFT 64 in a common leg. The dual path equalizer 60 also includes a first finite filter 66 and a first post-processor 68 in a first path 70, and a second finite filter 72 and a second post-processor 74 in a second path 76. The outputs of the first and second post-processors 68 and 74 are added by an adder 78, and the

output of the adder 78 is down sampled by two by a down sampler 79.

As can be seen by comparing Figures 5 and 7, the hardware of the dual path equalizer 60 is quite similar to the hardware of the dual path equalizer 40. However, the coefficients applied by the pre-processor 62, by the first and second finite filters 66 and 72, and by the first and second post-processors 68 and 74 are different.

The coefficients b_0 applied by the pre-processor 62 are curved as can be seen in Figure 7. The coefficients b_0 are curved according to the function $1/(2 - \cos(t))$. The coefficients b_0 applied by the pre-processor 62 are curved in order to minimize the complexity of the coefficients C_1 and C_2 applied by the first and second post-processors 48 and 54 of the dual path equalizer 40 shown in Figure 5. This curvature of the coefficients b_0 permits implementation of the first and second post-processors 68 and 74 as simplified two tap convolvers. Each tap of each convolver has a magnitude of one which permits the first and second post-processors 68 and 74 to be implemented as adders rather than as multipliers. As in the case of the prior embodiments, the width of (i.e., the number of coefficients

in) the set of coefficients \mathbf{b}_0 is varied as the interval d between the data block and its ghost varies. As the width of the set of coefficients \mathbf{b}_0 varies, the curvature of the coefficients \mathbf{b}_0 still follows the function $1/(2 - \cos(t))$.

5 The coefficients \mathbf{A}_1 and \mathbf{A}_2 applied respectively by the first and second finite filters 66 and 72 have non-zero real and imaginary parts. As discussed below, the coefficients \mathbf{A}_1 and \mathbf{A}_2 are adjusted during operation of the dual path equalizer 60 so that, when the outputs of the first and second paths 70 and 76 are added by the adder 78, ghosts are substantially eliminated and noise enhancement is substantially minimized.

10 The coefficients \mathbf{C}_1 applied by the first post-processor 68 are such that application of an Inverse Fast Fourier Transform to the coefficients \mathbf{C}_1 results in a set of coefficients in the time domain that have a curvature corresponding to the curvature of the coefficients \mathbf{b}_0 .
15 Similarly, the coefficients \mathbf{C}_2 applied by the second post-processor 74 are such that application of an Inverse Fast Fourier Transform to the coefficients \mathbf{C}_2 results in a set of
20 coefficients in the time domain that have a curvature corresponding to the curvature of the coefficients \mathbf{b}_0 .

It may be noted that the right most tap of the coefficients C_1 and the left most tap of the coefficients C_2 coincide. Accordingly, the coefficients C_1 and C_2 may be de-composed into three sets of coefficients and may be applied by three corresponding post-processors as shown in Figure 8. Figure 8 shows a fourth equalizer embodiment in the form of a triple path equalizer 80 which includes a pre-processor 82 and a 2xFFT 84 in a common leg. The triple path equalizer 80 also includes a first finite filter 86 and a first post-processor 88 in a first path 90, a second finite filter 92 and a second post-processor 94 in a second path 96, and a third finite filter 98 and a third post-processor 100 in a third path 102. The outputs of the first, second, and third post-processors 88, 94, and 100 are added by an adder 104. A down sampler 106 down samples by two the output of the adder 104.

The coefficients b_0 applied by the pre-processor 82 are the same as the coefficients b_0 applied by the pre-processor 62 of the dual path equalizer 60. The coefficients A_1 applied by the first finite filter 86, the coefficients A_2 applied by the second finite filter 92, and the coefficients A_3 applied by the third finite filter 98

are adjusted during operation of the triple path equalizer 80 so that, when the outputs of the first, second, and third paths 90, 96, and 102 are added by the adder 104, ghosts are substantially eliminated and noise enhancement is

5 substantially minimized. The coefficient C_1 applied by the first post-processor 88 is the left-hand value of the coefficients C_1 applied by the first post-processor 68, the coefficient C_2 applied by the second post-processor 94 is the common value of the coefficients C_1 and C_2 applied by the first and second post-processors 68 and 74, and the coefficient C_3 applied by the third post-processor 100 is the right-hand value of the coefficients C_2 applied by the second post-processor 74.

10 The single tap of the first post-processor 88 results in a shift of one sample to the left of the data processed in the first path 90, the single tap of the second post-processor 94 results in no sample shift of the data processed in the second path 96, and the single tap of the third post-processor 100 results in a shift of one sample to the right of the data processed in the third path 102.

20 Therefore, as shown by a triple path equalizer 110 of Figure

9, the first, second, and third post-processors 88, 94, and 100 can be replaced by sample shifters.

Accordingly, the triple path equalizer 110 includes a pre-processor 112 and a 2xFFT 114 in a common leg. The triple path equalizer 110 also includes a left one-sample shifter 116, a first by-two down sampler 118, and a first finite filter 120 in a first path 122, a second by-two down sampler 124 and a second finite filter 126 in a second path 128, and a right one-sample shifter 130, a third by-two down sampler 132, and a third finite filter 134 in a third path 136. The outputs of the first, second, and third finite filters 120, 126, and 134 are added by an adder 138.

The coefficients \mathbf{b}_0 applied by the pre-processor 112 are the same as the coefficients \mathbf{b}_0 applied by the pre-processor 82 of the triple path equalizer 80. The coefficients \mathbf{A}_1 applied by the first finite filter 120, the coefficients \mathbf{A}_2 applied by the second finite filter 126, and the coefficients \mathbf{A}_3 applied by the third finite filter 134 are adjusted during operation of the triple path equalizer 110 so that, when the outputs of the first, second, and third paths 122, 128, and 136 are added by the adder 138,

ghosts are substantially eliminated and noise enhancement is substantially minimized.

Figure 10 illustrates a flow graph for the triple path equalizer 110 shown in Figure 9. The variable n designates the number of the sample emerging from the 2xFFT 114. To produce an output sample D_n from the adder 138, the first finite filter 120 processes a sample H_{2n-1} , the second finite filter 126 processes a sample H_{2n} , and the third finite filter 134 processes a sample H_{2n+1} . Because of the second by-two down sampler 124, the second finite filter 126 processes only the even samples of the samples supplied by the 2xFFT 114. Because of the left one-sample shifter 116 and the first by-two down sampler 118, the first finite filter 120 processes the sample H_{2n-1} which is one odd sample behind the sample H_{2n} . Because of the right one-sample shifter 130 and the third by-two down sampler 132, the third finite filter 134 processes the sample H_{2n+1} which is one odd sample ahead of the sample H_{2n} .

Exemplary sets of the coefficients b_0 , A_1 , A_2 , and A_3 , are shown in Figure 11 assuming an interval d between a data block and its ghost of one sample (i.e., $1/16$ of a data block containing sixteen samples). The first column of

Figure 6 shows the real parts of the coefficients b_0 , A_1 , A_2 , and A_3 , the second column of Figure 11 shows the imaginary parts of the coefficients b_0 , A_1 , A_2 , and A_3 , and the third column of Figure 11 shows the absolute values of the coefficients b_0 , A_1 , A_2 , and A_3 .

The coefficients b_0 applied by the pre-processor 112 are the same as the coefficients b_0 applied by the pre-processor 62. As in the case of the triple path equalizer 80, the width of (i.e., the number of coefficients in) the set of coefficients b_0 is varied as the interval d between the data block and its ghost varies. As the width of the set of coefficients b_0 varies, the curvature of the coefficients b_0 still follows the function $1/(2 - \cos(t))$.

As can be seen from Figure 11, and as discussed above, the coefficients b_0 applied by the pre-processor 112 have a real part shaped according to the function $1/(2 - \cos(t))$ and a zero imaginary part and are arranged as a window function having a duration that is substantially coextensive with a received data block and its ghost. The coefficients A_1 applied by the first finite filter 120, the coefficients A_2 applied by the second finite filter 126, and

the coefficients A_3 applied by the third finite filter 134 are adjusted during operation of the triple path equalizer 110 so that, when the outputs of the first, second, and third paths 122, 128, and 136 are added by the adder 138, ghosts are substantially eliminated and noise enhancement is substantially minimized.

The triple path equalizer 110 shown in Figure 9 can be generalized by expanding the number of paths. Accordingly, a multipath equalizer 140 shown in Figure 12 includes a pre-processor 142 and a 2xFFT 144 in a common leg. The multipath equalizer 140 also includes a plurality of paths ... 146_{n-2} , 146_{n-1} , 146_n , 146_{n+1} , 146_{n+2} ... having corresponding sample shifters ... 148_{n-2} , 148_{n-1} , 148_{n+1} , 148_{n+2} ..., corresponding by-two down samplers ... 150_{n-2} , 150_{n-1} , 150_n , 150_{n+1} , 150_{n+2} ..., and corresponding finite filters ... 152_{n-2} , 152_{n-1} , 152_n , 152_{n+1} , 152_{n+2} The outputs of the finite filters ... 152_{n-2} , 152_{n-1} , 152_n , 152_{n+1} , 152_{n+2} ... are added by an adder 154.

The coefficients b_0 applied by the pre-processor 142 are the same as the coefficients b_0 applied by the pre-processor 82 of the triple path equalizer 80. As in the case of the triple path equalizer 80, the width of (i.e.,

the number of coefficients in) the set of coefficients b_0 is varied as the interval d between the data block and its ghost varies. As noted above, the window function applied by the pre-processor 142 eliminates any energy, primarily noise, that is outside of the data blocks and their ghosts. The pre-processor 142 can be eliminated. However, if the pre-processor 142 is eliminated, the finite filters ... 152_{n-2} , 152_{n-1} , 152_n , 152_{n+1} , 152_{n+2} ... will process more noise than would otherwise be the case. Thus, without the pre-processor 142, additional noise may be present in the output of the adder 154.

The coefficients ... A_{n-2} , A_{n-1} , A_n , A_{n+1} , A_{n+2} ... applied by the corresponding finite filters ... 152_{n-2} , 152_{n-1} , 152_n , 152_{n+1} , 152_{n+2} ... are adjusted during operation of the multipath equalizer 140 so that, when the outputs of the paths ... 146_{n-2} , 146_{n-1} , 146_n , 146_{n+1} , 146_{n+2} ... are added by the adder 154, ghosts are substantially eliminated and noise enhancement is substantially minimized.

Similarly, the dual path equalizer 40 as shown in Figure 5 can be generalized by expanding the number of paths. Accordingly, a multipath equalizer 160 shown in Figure 13 includes a pre-processor 162 and a 2xFFT 164 in a

common leg. The multipath equalizer 160 also includes
 finite filters ... 166_{n-2} , 166_{n-1} , 166_n , 166_{n+1} , 166_{n+2} ... and
 corresponding post-processors ... 168_{n-2} , 168_{n-1} , 168_n ,
 168_{n+1} , 168_{n+2} ... arranged in corresponding paths ... 170_{n-2} ,
 170_{n-1} , 170_n , 170_{n+1} , 170_{n+2} The outputs of the post-
 processors ... 168_{n-2} , 168_{n-1} , 168_n , 168_{n+1} , 168_{n+2} ... are
 added by an adder 172, and the output of the adder 172 is
 down sampled by two by a down sampler 174.

The finite filters ... 166_{n-2} , 166_{n-1} , 166_n , 166_{n+1} ,
 166_{n+2} ... may be implemented as complex multipliers that
 complex multiply the coefficients ... A_{n-2} , A_{n-1} , A_n , A_{n+1} ,
 A_{n+2} ... by the frequency domain output of the 2xFFT 164.
 The post-processors ... 168_{n-2} , 168_{n-1} , 168_n , 168_{n+1} , 168_{n+2}
 ... convolve the frequency domain outputs from the finite
 filters ... 166_{n-2} , 166_{n-1} , 166_n , 166_{n+1} , 166_{n+2} ... 46 with
 coefficients ... C_{n-2} , C_{n-1} , C_n , C_{n+1} , C_{n+2}

The coefficients ... A_{n-2} , A_{n-1} , A_n , A_{n+1} , A_{n+2} ...
 applied by the corresponding finite filters ... 166_{n-2} , 166_{n-1} ,
 166_n , 166_{n+1} , 166_{n+2} ... are adjusted during operation of
 the multipath equalizer 160 so that, when the outputs of the
 paths ... 170_{n-2} , 170_{n-1} , 170_n , 170_{n+1} , 170_{n+2} ... are added by

the adder 172, ghosts are substantially eliminated and noise enhancement is substantially minimized.

The coefficients ... C_{n-2} , C_{n-1} , C_n , C_{n+1} , C_{n+2} ... may include the coefficients C_1 and C_2 applied by the first and second post-processors 48 and 54 of the dual path equalizer 40 and additional coefficients selected to achieve a desired level of performance. Also, the pre-processor 162 can be eliminated. However, if the pre-processor 162 is eliminated, the finite filters ... 166_{n-2} , 166_{n-1} , 166_n , 166_{n+1} , 166_{n+2} ... will process more noise than would otherwise be the case. Thus, without the pre-processor 162, additional noise may be present in the output of the adder 172.

One of the advantages of the multipath equalizers 140 and 160 is that, by increasing the number of finite filters, any weighting functions characterizing the pre-processors 142 and 162 can be transferred to the finite filters. Accordingly, the pre-processors 142 and 162 can be eliminated. Alternatively, the weighting of the coefficients of the pre-processors 142 and 162 can be eliminated so that these coefficients form pure window functions.

CONVERGENCE

As discussed above, the **A** coefficients may be adaptively controlled by a converger 7 to ensure that the actual output of the equalizer converges on the correct output, i.e., an output free of ghosts. A first embodiment of the converger 7, i.e., an adaptive coefficient control 180, may be provided for the dual path equalizer 10 and is shown in Figure 14. The adaptive convergence control 180 includes a conjugater 182 which conjugates the data from the output of the 2xFFT 14 to facilitate the use of an LMS algorithm to converge the **A** coefficients of the equalizer 10. The data exiting the 2xFFT 14 is complex data. The conjugater 182 conjugates this data by reversing the sign of the imaginary part of the data.

The output of the conjugater 182 is supplied to first and second correlators 184 and 186. An error generator 188, which is discussed more fully below, generates an error based upon the output data from the dual path equalizer 10. This error must be processed in the same manner as the output of the finite filters 16 and 24 of the dual path equalizer 10. Therefore, the adaptive coefficient

control 180 includes an inverse first post-processor 190 and a 2xFFT 192. Also, the adaptive coefficient control 180 includes an inverse second post-processor 194 and a 2xFFT 196. The inverse first post-processor 190 and the inverse second post-processor 194 produce the inverse of the first and second post-processors 20 and 28. Additionally, the 2xFFT 192 and the 2xFFT 196 produce the inverse of the first and second 2xFFT⁻¹ 18 and 2xFFT⁻¹ 26 of the dual path equalizer 10. The outputs of the first and second 2xFFTs 192 and 196 are supplied to the corresponding correlators 184 and 186.

The correlators 184 and 186 multiply (i) the error from the error generator 188 as processed by the respective inverse first and second post-processors 190 and 194 and the respective first and second 2xFFTs 192 and 196 and (ii) the conjugated output of the 2xFFT 14. This multiplication effects a modified LMS algorithm to produce adjustments values for the **A** coefficients to converge the equalizer 10. In other words, the outputs of the correlators 184 and 186 are adjustment values of the coefficients **A**₁ and **A**₂ that, when added to the existing coefficients **A**₁ and **A**₂ applied by

the dual path equalizer 10, would cause the dual path equalizer 10 to eliminate ghosts from the received signal.

However, instead of correcting the coefficients A_1 and A_2 in one operation, the coefficients A_1 and A_2 are adjusted in increments. Therefore, multipliers 198 and 200 multiply the outputs of the correlators 184 and 186 by a quantity α , which has a value of less than one. The output of the multiplier 198 is added to the existing coefficients A_1 of the first finite filter 16, and the output of the multiplier 200 is added to the existing coefficients A_2 of the second finite filter 28. The value α is used so that these coefficients A_1 and A_2 are corrected in small increments in order to ensure a smooth convergence.

A second embodiment of the converger 7, i.e., an adaptive coefficient control 210, may be provided for the dual path equalizer 40 and is shown in Figure 15. The adaptive convergence control 210 includes a conjugater 212 which conjugates the data from the 2xFFT 44. The output of the conjugater 212 is supplied to first and second correlators 214 and 216. An error generator 218, which is shown in exemplary form in Figure 18, generates an error based upon the output data of the dual path equalizer 40.

10/20/90 04:00 PM
5 This error must be processed in the same manner as the
output of the finite filters 46 and 52 of the dual path
equalizer 40. Therefore, the adaptive coefficient control
210 includes an up sampler 220 which up samples by two the
output of the error generator 218 in order to thereby
reverse the effects of the down sampler 59. The adaptive
coefficient control 210 also includes an inverse first post-
processor 222 and an inverse second post-processor 224. The
inverse first post-processor 222 and the inverse second
post-processor 224 produce an inverse of the first and
second post-processors 48 and 54. The outputs of the
inverse first and second post-processors 222 and 224 are
supplied to the corresponding correlators 214 and 216.

10/20/90 04:00 PM
20 The correlators 214 and 216 multiply (i) the up
sampled error from the error generator 218 as processed by
the respective inverse first and second post-processors 222
and 224 and (ii) the conjugated reference data. In effect,
the outputs of the correlators 214 and 216 are adjustment
values of the coefficients A_1 and A_2 that, when added to the
existing coefficients A_1 and A_2 applied by the dual path
equalizer 40, would cause the dual path equalizer 40 to
eliminate ghosts from the received signal.

Again, instead of correcting the coefficients A_1 and A_2 with only one operation, the coefficients A_1 and A_2 are adjusted in increments. Therefore, multipliers 226 and 228 multiply the corresponding outputs of the correlators 214 and 216 by a quantity α , which has a value of less than one. The output of the multiplier 226 is added to coefficients A_1 , and the output of the multiplier 228 is added to the coefficients A_2 . The value α is used so that the coefficients A_1 and A_2 are corrected in small increments in order to ensure a smooth convergence.

A third embodiment (not shown) of the converger 7 may be provided for the dual path equalizer 60. In this case, the adaptive coefficient control according to the third embodiment would have the same general arrangement as the adaptive coefficient control 210 shown in Figure 15. However, the inverse first and second post-processors in this third embodiment would be different from the inverse first and second post-processors 222 and 224 of the adaptive coefficient control 210 because of the difference between the first and second post-processors 48 and 54 of the dual path equalizer 40 and the first and second post-processors 68 and 74 of the dual path equalizer 60.

5 A fourth embodiment of the converger 7, i.e., an
adaptive coefficient control 240, may be provided for the
triple path equalizer 80 and is shown in Figure 16. The
adaptive convergence control 240 includes a conjugater 242
which conjugates the data from the 2xFFT 84. The output of
the conjugater 242 is supplied to first, second, and third
correlators 244, 246, and 248. An error generator 250
generates an error based upon the output of the triple path
equalizer 80. The adaptive convergence control 240 includes
an up sampler 252 which up samples by two the output of the
error generator 250 in order to thereby reverse the effects
of the down sampler 106. The adaptive coefficient control
240 also includes an inverse first post-processor 254, an
inverse second post-processor 256, and an inverse third
post-processor 258. The inverse first, second, and third
post-processors 254, 256, and 258 produce an inverse of the
first, second, and third post-processors 88, 94, and 100 of
the triple path equalizer 80. The outputs of the inverse
first, second, and third post-processors 254, 256, and 258
are supplied to the corresponding correlators 244, 246, and
248.

5 The correlators 244, 246, and 248 multiply (i) the up sampled error from the error generator 250 as processed by the respective inverse first, second, and third post-processors 254, 256, and 258 and (ii) the conjugated output of the 2xFFT 84. In effect, the outputs of the correlators 244, 246, and 248 are adjustment values of the coefficients A_1 , A_2 , and A_3 that, when added to the existing coefficients A_1 , A_2 , and A_3 applied by the triple path equalizer 80, would cause the triple path equalizer 80 to substantially eliminate ghosts from the received signal.

10 Multipliers 260, 262, and 264 multiply the outputs of the correlators 244, 246, and 248 by a quantity α , which has a value of less than one. The output of the multiplier 260 is added to coefficients A_1 , the output of the multiplier 262 is added to the coefficients A_2 , and the output of the multiplier 264 is added to the coefficients A_3 .

15 A fifth embodiment of the converger 7, i.e., an adaptive coefficient control 270, may be provided for the triple path equalizer 110 and is shown in Figure 17. The adaptive convergence control 270 includes a conjugater 272 which conjugates the data from the 2xFFT 114. The output of

20

the conjugater 272 is processed along first, second, and third control paths 274, 276, and 278 which correspond to the first, second, and third paths 122, 128, and 136 of the triple path equalizer 110. The first control path 274 includes a left one-sample shifter 280, a first by-two down sampler 282, and a first correlator 284. The left one-sample shifter 280 and the first by-two down sampler 282 replicate the processing of the left one-sample shifter 116 and the first by-two down sampler 118 in the first path 122 of the triple path equalizer 110. The second control path 276 includes a second by-two down sampler 286 and a second correlator 288. The second by-two down sampler 286 replicates the processing of the second by-two down sampler 124 in the second path 128 of the triple path equalizer 110. The third control path 278 includes a right one-sample shifter 290, a third by-two down sampler 292, and a third correlator 294. The right one-sample shifter 290 and the third by-two down sampler 292 replicate the processing of the right one-sample shifter 130 and the third by-two down sampler 132 in the third path 136 of the triple path equalizer 110. An error generator 296 generates an error based upon the output of the triple path equalizer 110.

5 The error from the error generator 296 is supplied
to the correlators 284, 288, and 294. The correlators 284,
288, and 294 multiply the error from the error generator 296
and the respective outputs of the first, second, and third
by-two down samplers 282, 286, and 292. In effect, the
outputs of the correlators 284, 288, and 294 are adjustment
values of the coefficients A_1 , A_2 , and A_3 that, when added
to the existing coefficients A_1 , A_2 , and A_3 applied by the
triple equalizer 110, would cause the triple path equalizer
110 to eliminate ghosts from the received signal.

10 Multipliers 298, 300, and 302 multiply the outputs
of the correlators 284, 288, and 294 by a quantity α , which
has a value of less than one. The output of the multiplier
298 is added to the existing coefficients A_1 , the output of
the multiplier 300 is added to the existing coefficients A_2 ,
15 and the output of the multiplier 302 is added to the
existing coefficients A_3 .

20 By the same token, the adaptive coefficient
control 270 can be generalized in order to converge the
output of the multipath equalizer 140 into a ghost free
signal, and the adaptive coefficient control 210 can be

generalized in order to converge the output of the multipath equalizer 160 into a ghost free signal.

5 An error generator 400 is shown in Figure 18 and may be used for the error generators 188, 218, 250, and 296. The adaptive coefficient controls of the present invention may be operated in either a training mode or a data directed mode. During the training mode, a training signal is transmitted by a transmitter to the corresponding equalizer. A summer 402 subtracts the training signal (which may be locally stored or generated) from the output of the equalizer (which is the received training signal) and supplies this difference to a switch 404. During the data directed mode, the data output of the equalizer is sliced by a slicer 406 using stored slice levels 408 in order to generate as an output the one stored slice level which is closest to the received data. The stored slice levels 408 are the reference in the data directed mode. A summer 410 subtracts the stored slice level provided by the slicer 406 from the output of the equalizer and supplies this difference to the switch 404. The switch 404 selects either the output of the summer 402 or the output from the summer 410 as the error which is processed by the particular

adaptive coefficient control according to the various descriptions above. The switch 404 may be operated by the synchronizer 6 shown in Figure 2.

5 The switch 404 may select the output of the summer 402 (i.e., the training mode) during system start up when the **A** coefficients do not have values likely to produce meaningful output data. Once the equalizer has been coarsely converged during the training mode, the switch 404 may be switched to select the output of the summer 410
10 (i.e., the data directed mode) so that data can be used in order to achieve and maintain precise convergence of the equalizer.

15 Certain modifications and alternatives of the present invention have been discussed above. Other modifications and alternatives will occur to those practicing in the art of the present invention. For example, because the present invention operates most satisfactorily in the presence of ghosts and other linear distortions, the term ghost as used herein in connection with the present
20 invention includes ghosts and/or other linear distortions.

Also, the Fast Fourier Transforms and inverse Fast Fourier Transforms described above can have lengths other than those described above.

Furthermore, the invention has been described above as if a single ghost is received. In the case where multiple ghosts are received, it may be desirable to apply multiple sets of the coefficients to the received signal or a single set tailored for multiple ghosts. Also, the spacing between coefficients is described above as being the interval d . However, in the case where the interval d is not evenly divisible into the block length of a data block, or in the case where more than one ghost are received, the spacing between the coefficients may be other than the interval d .

The guard interval discussed above may have any desired values including zero.

Moreover, as described above, the error generated by the error generator 400 is correlated with the conjugated output of a 2xFFT that is part of the corresponding equalizer, and this correlation is used to adjust the appropriate **A** coefficients. Instead, the error generated by the error generator 400 could be used in a zero forcing

method to directly adjust the appropriate **A** coefficients without correlation with the conjugated output of the corresponding 2xFFT. The zero forcing method, however, may not suitably account for noise in the input.

5

Accordingly, the description of the present invention is to be construed as illustrative only and is for the purpose of teaching those skilled in the art the best mode of carrying out the invention. The details may be varied substantially without departing from the spirit of the invention, and the exclusive use of all modifications which are within the scope of the appended claims is reserved.

0040700
10
0040700

WHAT IS CLAIMED IS:

1 1. A method of equalizing a signal, wherein the
2 signal comprises a series of input blocks of coded data, the
3 method comprising at least the following steps:

4 a) shifting data in each input block of data to
5 the left;

6 b) complex multiplying each of the left shifted
7 input block of data by a first set of equalizer coefficients
8 to provide respective first adjusted output blocks of data,
9 wherein step b) is not a full solution to ghosts;

10 c) complex multiplying each of the input blocks of
11 data by a second set of equalizer coefficients to provide
12 respective second adjusted output blocks of data, wherein
13 step c) is not a full solution to ghosts;

14 d) shifting the data in each input block of data
15 to the right;

16 e) complex multiplying each of the right shifted
17 input block of data by a third set of equalizer coefficients
18 to provide respective third adjusted output blocks of data,
19 wherein step e) is not a full solution to ghosts;

20 f) adding corresponding ones of the first, second,
 21 and third adjusted output blocks of data; and,
 22 g) controlling the first, second, and third sets
 23 of equalizer coefficients so that, as a result of the
 24 addition performed according to step f), a substantially
 25 full solution to ghosts is obtained.

1 2. The method of claim 1 wherein step g)
 2 comprises the following steps:
 3 estimating the channel; and,
 4 controlling the first, second, and third sets of
 5 equalizer coefficients based upon the estimated channel.

6 3. The method of claim 1 wherein step g)
 7 comprises the step of controlling the first, second, and
 8 third sets of equalizer coefficients based upon the addition
 9 of step f).

1 4. The method of claim 1 wherein step g)
 2 comprises the following steps:
 3 g1) performing a comparison based upon the
 4 addition of step f) and the input blocks of data; and,

5 g2) controlling the first, second, and third sets
6 of equalizer coefficients based upon the comparison
7 performed in step g1).

1 5. The method of claim 1 wherein step g)
2 comprises the following steps:

3 g1) comparing results of the addition of step f)
4 to a reference to form an error;

5 g2) conjugating the input blocks of data;

6 g3) shifting data in the conjugated input blocks
7 of data to the left;

8 g4) shifting the data in the conjugated input
9 blocks of data to the right;

10 g5) correlating the left shifted, conjugated input
11 blocks of data and the error;

12 g6) correlating the conjugated input blocks of
13 data and the error;

14 g7) correlating the right shifted, conjugated
15 input blocks of data and the error;

16 g8) controlling the first set of equalizer
17 coefficients based upon the correlation performed at step
18 g5);

19 g9) controlling the second set of equalizer
20 coefficients based upon the correlation performed at step
21 g6); and,
22 g10) controlling the third set of equalizer
23 coefficients based upon the correlation performed at step
24 g7).

1 6. The method of claim 5 wherein step g5)
2 comprises the step of down sampling the left shifted,
3 conjugated input blocks of data, wherein step g6) comprises
4 the step of down sampling the conjugated input blocks of
5 data, and wherein step g7) comprises the step of down
6 sampling the right shifted, conjugated input blocks of data.

7 7. The method of claim 5 wherein the reference is
8 a training signal.

1 8. The method of claim 5 wherein the reference is
2 sliced data.

1 9. The method of claim 1 wherein step g)
2 comprises the following steps:
3 g1) comparing results of the addition of step f)
4 to a reference to form an error;
5 g2) performing a left shift based upon the input
6 blocks of data;
7 g3) performing a right shift based upon the input
8 blocks of data;
9 g4) performing a first correlation based upon
10 results from step g2) and the error;
11 g5) performing a second correlation based upon the
12 input blocks of data and the error;
13 g6) performing a third correlation based upon
14 results from step g3) and the error;
15 g7) controlling the first set of equalizer
16 coefficients based upon the first correlation;
17 g8) controlling the second set of equalizer
18 coefficients based upon the second correlation; and,
19 g9) controlling the third set of equalizer
20 coefficients based upon the third correlation.

1 10. The method of claim 11 wherein the reference
2 is a training signal.

1 11. The method of claim 11 wherein the reference
2 is sliced data.

1 12. The method of claim 1 wherein step g)
2 comprises the following steps:

3 g1) conjugating the input blocks of data;

4 g2) shifting data in the conjugated input blocks
5 of data to the left;

6 g3) shifting the data in the conjugated input
7 blocks of data to the right;

8 g4) performing a first correlation based upon the
9 left shifted, conjugated input blocks of data and the
10 addition of step f);

11 g5) performing a second correlation based upon the
12 conjugated input blocks of data and the addition of step f);

13 g6) performing a third correlation based upon the
14 right shifted, conjugated input blocks of data and the
15 addition of step f);

16 g7) controlling the first set of equalizer
17 coefficients based upon the first correlation;
18 g8) controlling the second set of equalizer
19 coefficients based upon the second correlation; and,
20 g9) controlling the third set of equalizer
21 coefficients based upon the third correlation.

1 13. The method of claim 1 wherein step g)
2 comprises the following steps:
3

4 g1) performing a left shift based upon the input
5 blocks of data;

6 g2) performing a right shift based upon the input
7 blocks of data;

8 g3) performing a first correlation based upon
9 results from step g1) and the first, second, and third
adjusted output blocks of data;

10 g4) performing a second correlation based upon the
11 input blocks of data and the first, second, and third
12 adjusted output blocks of data;

13 g5) performing a third correlation based upon
14 results from step g2) and the first, second, and third
15 adjusted output blocks of data;

16 g6) controlling the first set of equalizer
17 coefficients based upon the first correlation;
18 g7) controlling the second set of equalizer
19 coefficients based upon the second correlation; and,
20 g8) controlling the third set of equalizer
21 coefficients based upon the third correlation.

1 14. The method of claim 1 further comprising the
2 step of h) applying pre-processing coefficients to each data
3 block prior to steps a), c), and d).
4
5
6

1 15. The method of claim 14 wherein step g)
2 comprises the step of controlling a width of the pre-
3 processing coefficients so that the width of the pre-
4 processing coefficients is substantially commensurate with a
5 width of a data block and an interval between a data block
6 and a ghost.

1 16. The method of claim 14 wherein step g)
2 comprises the following steps:
3 estimating the channel; and,
4 controlling the first, second, and third sets of
5 equalizer coefficients based upon the estimated channel.

1 17. The method of claim 14 wherein step g)
2 comprises the step of controlling the first, second, and
3 third sets of equalizer coefficients based upon the addition
4 of step f).
5

1 18. The method of claim 14 wherein step g)
2 comprises the following steps:
3 g1) performing a comparison based upon the
4 addition of step f) and the input blocks of data; and,
5 g2) controlling the first, second, and third sets
6 of equalizer coefficients based upon the comparison
7 performed in step g1).

6 g3) shifting data in the conjugated input blocks
 7 of data to the left;
 8 g4) shifting the data in the conjugated input
 9 blocks of data to the right;
 10 g5) correlating the left shifted, conjugated input
 11 blocks of data and the error;
 12 g6) correlating the conjugated input blocks of
 13 data and the error;
 14 g7) correlating the right shifted, conjugated
 15 input blocks of data and the error;
 16 g8) controlling the first set of equalizer
 17 coefficients based upon the correlation performed at step
 18 g5);
 19 g9) controlling the second set of equalizer
 20 coefficients based upon the correlation performed at step
 21 g6); and,
 22 g10) controlling the third set of equalizer
 23 coefficients based upon the correlation performed at step
 24 g7).

1 23. The method of claim 22 wherein step g5)
2 comprises the step of down sampling the left shifted,
3 conjugated input blocks of data, wherein step g6) comprises
4 the step of down sampling the conjugated input blocks of
5 data, and wherein step g7) comprises the step of down
6 sampling the right shifted, conjugated input blocks of data.

1 24. The method of claim 22 wherein the reference
2 is a training signal.

3 25. The method of claim 22 wherein the reference
4 is sliced data.

5 26. The method of claim 14 wherein step g)
6 comprises the following steps:

7 g1) comparing results of the addition of step f)
8 to reference data to form an error;

9 g2) performing a left shift based upon the input
10 blocks of data;

11 g3) performing a right shift based upon the input
12 blocks of data;

9 g4) performing a first correlation based upon
10 results from step g2) and the error;

11 g5) performing a second correlation based upon the
12 input blocks of data and the error;

13 g6) performing a third correlation based upon
14 results from step g3) and the error;

15 g7) controlling the first set of equalizer
16 coefficients based upon the first correlation;

17 g8) controlling the second set of equalizer
18 coefficients based upon the second correlation; and,

19 g9) controlling the third set of equalizer
20 coefficients based upon the third correlation.

21 27. The method of claim 26 wherein the reference
22 is a training signal.

23 28. The method of claim 26 wherein the reference
24 is sliced data.

1 29. The method of claim 21 wherein step g)
2 comprises the following steps:
3 g1) conjugating the input blocks of data;
4 g2) shifting data in the conjugated input blocks
5 of data to the left;
6 g3) shifting the data in the conjugated input
7 blocks of data to the right;
8 g4) performing a first correlation based upon the
9 left shifted, conjugated input blocks of data and the
10 addition of step f);
11 g5) performing a second correlation based upon the
12 conjugated input blocks of data and the addition of step f);
13 g6) performing a third correlation based upon the
14 right shifted, conjugated input blocks of data and the
15 addition of step f);
16 g7) controlling the first set of equalizer
17 coefficients based upon the first correlation;
18 g8) controlling the second set of equalizer
19 coefficients based upon the second correlation; and,
20 g9) controlling the third set of equalizer
21 coefficients based upon the third correlation.

1 30. The method of claim 14 wherein step g)
2 comprises the following steps:
3 g1) performing a left shift based upon the input
4 blocks of data;
5 g2) performing a right shift based upon the input
6 blocks of data;
7 g3) performing a first correlation based upon
8 results from step g1) and the first, second, and third
9 adjusted output blocks of data;
10 g4) performing a second correlation based upon the
11 input blocks of data and the first, second, and third
12 adjusted output blocks of data;
13 g5) performing a third correlation based upon
14 results from step g2) and the first, second, and third
15 adjusted output blocks of data;
16 g6) controlling the first set of equalizer
17 coefficients based upon the first correlation;
18 g7) controlling the second set of equalizer
19 coefficients based upon the second correlation; and,
20 g8) controlling the third set of equalizer
21 coefficients based upon the third correlation.

1 31. The method of claim 14 wherein the pre-
2 processing coefficients are curved.

1 32. The method of claim 14 wherein the pre-
2 processing coefficients are curved substantially according
3 to a function $1/(2 - \cos(t))$.

1 33. The method of claim 1 further comprising the
2 step of applying a spectral transformation to each data
3 block prior to steps a), c), and d), wherein the spectral
4 transformation is longer than a data block.

1 34. An equalizer for processing blocks of data
2 comprising:

3 n - 1 data shifters, wherein each of the n - 1
4 data shifters shifts the blocks of data;

5 n finite filters, wherein one of the n finite
6 filters applies a corresponding set of finite filter
7 coefficients to the blocks of data, wherein each of the
8 other n - 1 finite filters applies a set of finite filter
9 coefficients to a corresponding output of the n - 1 data
10 shifters, wherein ghosts of the blocks of data are not

11 eliminated as a result of the application of the sets of
12 finite filter coefficients corresponding to the n finite
13 filters, and wherein $n > 2$;

14 an adder arranged to add outputs from the n finite
15 filters; and,

16 a controller arranged to control the sets of
17 finite filter coefficients corresponding to the n finite
18 filters so that the addition performed by the adder
19 substantially eliminates the ghosts.

20 35. The equalizer of claim 34 wherein the
21 controller comprises a channel estimator that estimates the
22 channel through which the blocks of data are transmitted.

23 36. The equalizer of claim 34 wherein the
24 controller controls the n sets of finite filter coefficients
25 based upon an output of the adder.

26 37. The equalizer of claim 34 wherein the
27 controller comprises a comparator that performs a comparison
28 based upon an output of the adder and the blocks of data.

1 38. The equalizer of claim 34 wherein the
2 controller comprises:

3 a comparator arranged to compare an output of the
4 adder to a reference to form an error;

5 a conjugator arranged to conjugate the blocks of
6 data;

7 n - 1 data shifters arranged to shift the
8 conjugated blocks of data; and,

9 n correlators arranged to perform n - 1
10 correlations of the shifted, conjugated blocks of data and
11 the error and to perform one correlation of the conjugated
12 blocks of data and the error, wherein each of the n
13 correlators is arranged to control a corresponding set of
14 equalizer coefficients.

1 39. The equalizer of claim 38 wherein the
2 controller further comprises n down samplers, wherein n - 1
3 of the n down samplers are arranged to down sample the
4 shifted, conjugated blocks of data prior upstream of the
5 correlators, and wherein the other down sampler is arranged
6 to down sample the conjugated blocks of data upstream of the
7 correlators.

1 40. The equalizer of claim 38 wherein the
2 reference is a training signal.

1 41. The equalizer of claim 38 wherein the
2 reference is sliced data.

1 42. The equalizer of claim 34 wherein the
2 controller comprises:

3 a comparator arranged to compare an output of the
4 adder to a reference to form an error;

5 n - 1 data shifters arranged to shift the blocks
6 of data; and,

7 n correlators arranged to perform n - 1
8 correlations based upon the shifted blocks of data and the
9 error and to perform one correlation based upon the blocks
10 of data and the error, wherein each of the n correlators is
11 arranged to control a corresponding set of equalizer
12 coefficients.

1 43. The equalizer of claim 42 wherein the
2 reference is a training signal.

1 44. The equalizer of claim 42 wherein the
2 reference is sliced data.

1 45. The equalizer of claim 34 wherein the
2 controller comprises:

3 a conjugator arranged to conjugate the blocks of
4 data;

5 n - 1 data shifters arranged to shift the
6 conjugated blocks of data;

7 n correlators arranged to perform n - 1
8 correlations based upon the shifted, conjugated blocks of
9 data and an output of the adder and to perform one
10 correlation based upon the conjugated blocks of data and the
11 output of the adder, wherein each of the n correlators is
12 arranged to control a corresponding set of equalizer
13 coefficients.

1 46. The equalizer of claim 34 wherein the
2 controller comprises:

3 n - 1 data shifters arranged to shift the blocks
4 of data; and,

5 n correlators arranged to perform $n - 1$
6 correlations based upon the shifted blocks of data and an
7 output of the adder and to perform one correlation based
8 upon the blocks of data and the output of the adder, wherein
9 each of the n correlators is arranged to control a
10 corresponding set of equalizer coefficients.

1 47. The equalizer of claim 34 further comprising
2 a pre-processor that applies pre-processor coefficients to
3 each data block upstream of the $n - 1$ data shifters and the
4 n finite filters.

5 48. The equalizer of claim 47 wherein the
6 controller controls a width of the pre-processing
7 coefficients so that the width is substantially coincident
8 with the width of a data block and an interval between a
9 data block and a ghost.

1 49. The equalizer of claim 47 wherein the pre-
2 processing coefficients are curved.

1 50. The equalizer of claim 47 wherein the pre-
2 processing coefficients are curved substantially according
3 to a function $1/(2 - \cos(t))$.

1 51. The equalizer of claim 47 wherein the
2 controller comprises a channel estimator that estimates the
3 channel through which the blocks of data are transmitted.

1 52. The equalizer of claim 47 wherein the
2 controller controls the n sets of finite filter coefficients
3 based upon an output of the adder.

1 53. The equalizer of claim 47 wherein the
2 controller comprises a comparator that performs a comparison
3 based upon an output of the adder and the blocks of data.

1 54. The equalizer of claim 47 wherein the
2 controller comprises:
3 a comparator arranged to compare an output of the
4 adder to a reference to form an error;
5 a conjugator arranged to conjugate the blocks of
6 data;

7 n - 1 data shifters arranged to shift the
8 conjugated blocks of data; and,
9 n correlators arranged to perform n - 1
10 correlations of the shifted, conjugated blocks of data and
11 the error and to perform one correlation of the conjugated
12 blocks of data and the error, wherein each of the n
13 correlators is arranged to control a corresponding set of
14 equalizer coefficients.

55. The equalizer of claim 54 wherein the
controller further comprises n down samplers, wherein n - 1
of the n down samplers are arranged to down sample the
shifted, conjugated blocks of data prior upstream of the
correlators, and wherein the other down sampler is arranged
to down sample the conjugated blocks of data upstream of the
correlators.

56. The equalizer of claim 54 wherein the
reference is a training signal.

57. The equalizer of claim 54 wherein the
reference is sliced data.

1 58. The equalizer of claim 47 wherein the
2 controller comprises:

3 a comparator arranged to compare an output of the
4 adder to a reference to form an error;

5 n - 1 data shifters arranged to shift the blocks
6 of data; and,

7 n correlators arranged to perform n - 1
8 correlations based upon the shifted blocks of data and the
9 error and to perform one correlation based upon the blocks
10 of data and the error, wherein each of the n correlators is
11 arranged to control a corresponding set of equalizer
12 coefficients.

13 59. The equalizer of claim 58 wherein the
14 reference is a training signal.

15 60. The equalizer of claim 58 wherein the
16 reference is sliced data.

1 61. The equalizer of claim 47 wherein the
2 controller comprises:

3 a conjugator arranged to conjugate the blocks of
4 data;

5 n - 1 data shifters arranged to shift the
6 conjugated blocks of data;

7 n correlators arranged to perform n - 1
8 correlations based upon the shifted, conjugated blocks of
9 data and an output of the adder and to perform one
10 correlation based upon the conjugated blocks of data and the
11 output of the adder, wherein each of the n correlators is
12 arranged to control a corresponding set of equalizer
13 coefficients.

14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
101
102
103
104
105
106
107
108
109
110
111
112
113
114
115
116
117
118
119
120
121
122
123
124
125
126
127
128
129
130
131
132
133
134
135
136
137
138
139
140
141
142
143
144
145
146
147
148
149
150
151
152
153
154
155
156
157
158
159
160
161
162
163
164
165
166
167
168
169
170
171
172
173
174
175
176
177
178
179
180
181
182
183
184
185
186
187
188
189
190
191
192
193
194
195
196
197
198
199
200
201
202
203
204
205
206
207
208
209
210
211
212
213
214
215
216
217
218
219
220
221
222
223
224
225
226
227
228
229
230
231
232
233
234
235
236
237
238
239
240
241
242
243
244
245
246
247
248
249
250
251
252
253
254
255
256
257
258
259
260
261
262
263
264
265
266
267
268
269
270
271
272
273
274
275
276
277
278
279
280
281
282
283
284
285
286
287
288
289
290
291
292
293
294
295
296
297
298
299
300
301
302
303
304
305
306
307
308
309
310
311
312
313
314
315
316
317
318
319
320
321
322
323
324
325
326
327
328
329
330
331
332
333
334
335
336
337
338
339
340
341
342
343
344
345
346
347
348
349
350
351
352
353
354
355
356
357
358
359
360
361
362
363
364
365
366
367
368
369
370
371
372
373
374
375
376
377
378
379
380
381
382
383
384
385
386
387
388
389
390
391
392
393
394
395
396
397
398
399
400
401
402
403
404
405
406
407
408
409
410
411
412
413
414
415
416
417
418
419
420
421
422
423
424
425
426
427
428
429
430
431
432
433
434
435
436
437
438
439
440
441
442
443
444
445
446
447
448
449
450
451
452
453
454
455
456
457
458
459
460
461
462
463
464
465
466
467
468
469
470
471
472
473
474
475
476
477
478
479
480
481
482
483
484
485
486
487
488
489
490
491
492
493
494
495
496
497
498
499
500
501
502
503
504
505
506
507
508
509
510
511
512
513
514
515
516
517
518
519
520
521
522
523
524
525
526
527
528
529
530
531
532
533
534
535
536
537
538
539
540
541
542
543
544
545
546
547
548
549
550
551
552
553
554
555
556
557
558
559
560
561
562
563
564
565
566
567
568
569
570
571
572
573
574
575
576
577
578
579
580
581
582
583
584
585
586
587
588
589
590
591
592
593
594
595
596
597
598
599
600
601
602
603
604
605
606
607
608
609
610
611
612
613
614
615
616
617
618
619
620
621
622
623
624
625
626
627
628
629
630
631
632
633
634
635
636
637
638
639
640
641
642
643
644
645
646
647
648
649
650
651
652
653
654
655
656
657
658
659
660
661
662
663
664
665
666
667
668
669
670
671
672
673
674
675
676
677
678
679
680
681
682
683
684
685
686
687
688
689
690
691
692
693
694
695
696
697
698
699
700
701
702
703
704
705
706
707
708
709
710
711
712
713
714
715
716
717
718
719
720
721
722
723
724
725
726
727
728
729
730
731
732
733
734
735
736
737
738
739
740
741
742
743
744
745
746
747
748
749
750
751
752
753
754
755
756
757
758
759
760
761
762
763
764
765
766
767
768
769
770
771
772
773
774
775
776
777
778
779
780
781
782
783
784
785
786
787
788
789
790
791
792
793
794
795
796
797
798
799
800
801
802
803
804
805
806
807
808
809
810
811
812
813
814
815
816
817
818
819
820
821
822
823
824
825
826
827
828
829
830
831
832
833
834
835
836
837
838
839
840
841
842
843
844
845
846
847
848
849
850
851
852
853
854
855
856
857
858
859
860
861
862
863
864
865
866
867
868
869
870
871
872
873
874
875
876
877
878
879
880
881
882
883
884
885
886
887
888
889
890
891
892
893
894
895
896
897
898
899
900
901
902
903
904
905
906
907
908
909
910
911
912
913
914
915
916
917
918
919
920
921
922
923
924
925
926
927
928
929
930
931
932
933
934
935
936
937
938
939
940
941
942
943
944
945
946
947
948
949
950
951
952
953
954
955
956
957
958
959
960
961
962
963
964
965
966
967
968
969
970
971
972
973
974
975
976
977
978
979
980
981
982
983
984
985
986
987
988
989
990
991
992
993
994
995
996
997
998
999
1000
1001
1002
1003
1004
1005
1006
1007
1008
1009
1010
1011
1012
1013
1014
1015
1016
1017
1018
1019
1020
1021
1022
1023
1024
1025
1026
1027
1028
1029
1030
1031
1032
1033
1034
1035
1036
1037
1038
1039
1040
1041
1042
1043
1044
1045
1046
1047
1048
1049
1050
1051
1052
1053
1054
1055
1056
1057
1058
1059
1060
1061
1062
1063
1064
1065
1066
1067
1068
1069
1070
1071
1072
1073
1074
1075
1076
1077
1078
1079
1080
1081
1082
1083
1084
1085
1086
1087
1088
1089
1090
1091
1092
1093
1094
1095
1096
1097
1098
1099
1100
1101
1102
1103
1104
1105
1106
1107
1108
1109
1110
1111
1112
1113
1114
1115
1116
1117
1118
1119
1120
1121
1122
1123
1124
1125
1126
1127
1128
1129
1130
1131
1132
1133
1134
1135
1136
1137
1138
1139
1140
1141
1142
1143
1144
1145
1146
1147
1148
1149
1150
1151
1152
1153
1154
1155
1156
1157
1158
1159
1160
1161
1162
1163
1164
1165
1166
1167
1168
1169
1170
1171
1172
1173
1174
1175
1176
1177
1178
1179
1180
1181
1182
1183
1184
1185
1186
1187
1188
1189
1190
1191
1192
1193
1194
1195
1196
1197
1198
1199
1200
1201
1202
1203
1204
1205
1206
1207
1208
1209
1210
1211
1212
1213
1214
1215
1216
1217
1218
1219
1220
1221
1222
1223
1224
1225
1226
1227
1228
1229
1230
1231
1232
1233
1234
1235
1236
1237
1238
1239
1240
1241
1242
1243
1244
1245
1246
1247
1248
1249
1250
1251
1252
1253
1254
1255
1256
1257
1258
1259
1260
1261
1262
1263
1264
1265
1266
1267
1268
1269
1270
1271
1272
1273
1274
1275
1276
1277
1278
1279
1280
1281
1282
1283
1284
1285
1286
1287
1288
1289
1290
1291
1292
1293
1294
1295
1296
1297
1298
1299
1300
1301
1302
1303
1304
1305
1306
1307
1308
1309
1310
1311
1312
1313
1314
1315
1316
1317
1318
1319
1320
1321
1322
1323
1324
1325
1326
1327
1328
1329
1330
1331
1332
1333
1334
1335
1336
1337
1338
1339
1340
1341
1342
1343
1344
1345
1346
1347
1348
1349
1350
1351
1352
1353
1354
1355
1356
1357
1358
1359
1360
1361
1362
1363
1364
1365
1366
1367
1368
1369
1370
1371
1372
1373
1374
1375
1376
1377
1378
1379
1380
1381
1382
1383
1384
1385
1386
1387
1388
1389
1390
1391
1392
1393
1394
1395
1396
1397
1398
1399
1400
1401
1402
1403
1404
1405
1406
1407
1408
1409
1410
1411
1412
1413
1414
1415
1416
1417
1418
1419
1420
1421
1422
1423
1424
1425
1426
1427
1428
1429
1430
1431
1432
1433
1434
1435
1436
1437
1438
1439
1440
1441
1442
1443
1444
1445
1446
1447
1448
1449
1450
1451
1452
1453
1454
1455
1456
1457
1458
1459
1460
1461
1462
1463
1464
1465
1466
1467
1468
1469
1470
1471
1472
1473
1474
1475
1476
1477
1478
1479
1480
1481
1482
1483
1484
1485
1486
1487
1488
1489
1490
1491
1492
1493
1494
1495
1496
1497
1498
1499
1500
1501
1502
1503
1504
1505
1506
1507
1508
1509
1510
1511
1512
1513
1514
1515
1516
1517
1518
1519
1520
1521
1522
1523
1524
1525
1526
1527
1528
1529
1530
1531
1532
1533
1534
1535
1536
1537
1538
1539
1540
1541
1542
1543
1544
1545
1546
1547
1548
1549
1550
1551
1552
1553
1554
1555
1556
1557
1558
1559
1560
1561
1562
1563
1564
1565
1566
1567
1568
1569
1570
1571
1572
1573
1574
1575
1576
1577
1578
1579
1580
1581
1582
1583
1584
1585
1586
1587
1588
1589
1590
1591
1592
1593
1594
1595
1596
1597
1598
1599
1600
1601
1602
1603
1604
1605
1606
1607
1608
1609
1610
1611
1612
1613
1614
1615
1616
1617
1618
1619
1620
1621
1622
1623
1624
1625
1626
1627
1628
1629
1630
1631
1632
1633
1634
1635
1636
1637
1638
1639
1640
1641
1642
1643
1644
1645
1646
1647
1648
1649
1650
1651
1652
1653
1654
1655
1656
1657
1658
1659
1660
1661
1662
1663
1664
1665
1666
1667
1668
1669
1670
1671
1672
1673
1674
1675
1676
1677
1678
1679
1680
1681
1682
1683
1684
1685
1686
1687
1688
1689
1690
1691
1692
1693
1694
1695
1696
1697
1698
1699
1700
1701
1702
1703
1704
1705
1706
1707
1708
1709
1710
1711
1712
1713
1714
1715
1716
1717
1718
1719
1720
1721
1722
1723
1724
1725
1726
1727
1728
1729
1730
1731
1732
1733
1734
1735
1736
1737
1738
1739
1740
1741
1742
1743
1744
1745
1746
1747
1748
1749
1750
1751
1752
1753
1754
1755
1756
1757
1758
1759
1760
1761
1762
1763
1764
1765
1766
1767
1768
1769
1770
1771
1772
1773
1774
1775
1776
1777
1778
1779
1780
1781
1782
1783
1784
1785
1786
1787
1788
1789
1790
1791
1792
1793
1794
1795
1796
1797
1798
1799
1800
1801
1802
1803
1804
1805
1806
1807
1808
1809
1810
1811
1812
1813
1814
1815
1816
1817
1818
1819
1820
1821
1822
1823
1824
1825
1826
1827
1828
1829
1830
1831
1832
1833
1834
1835
1836
1837
1838
1839
1840
1841
1842
1843
1844
1845
1846
1847
1848
1849
1850
1851
1852
1853
1854
1855
1856
1857
1858
1859
1860
1861
1862
1863
1864
1865
1866
1867
1868
1869
1870
1871
1872
1873
1874
1875
1876
1877
1878
1879
1880
1881
1882
1883
1884
1885
1886
1887
1888
1889
1890
1891
1892
1893
1894
1895
1896
1897
1898
1899
1900
1901
1902
1903
1904
1905
1906
1907
1908
1909
1910
1911
1912
1913
1914
1915
1916
1917
1918
1919
1920
1921
1922
1923
1924
1925
1926
1927
1928
1929
1930
1931
1932
1933
1934
1935
1936
1937
1938
1939
1940
1941
1942
1943
1944
1945
1946
1947
1948
1949
1950
1951
1952
1953
1954
1955
1956
1957
1958
1959
1960
1961
1962
1963
1964
1965
1966
1967
1968
1969
1970
1971
1972
1973
1974
1975
1976
1977
1978
1979
1980
1981
1982
1983
1984
1985
1986
1987
1988
1989
1990
1991
1992
1993
1994
1995
1996
1997
1998
1999
2000
2001
2002
2003
2004
2005
2006
2007
2008
2009
2010
2011
2012
2013
2014
2015
2016
2017
2018
2019
2020
2021
2022
2023
2024
2025
2026
2027
2028
2029
2030
2031
2032
2033
2034
2035
2036
2037
2038
2039
2040
2041
2042
2043
2044
2045
2046
2047
2048
2049
2050
2051
2052
2053
2054
2055
2056
2057
2058
2059
2060
2061
2062
2063
2064
2065
2066
2067
2068
2069
2070
2071
2072
2073
2074
2075
2076
2077
2078
2079
2080
2081
2082
2083
2084
2085
2086
2087
2088
2089
2090
2091
2092
2093
2094
2095
2096
2097
2098
2099
2100
2101
2102
2103
2104
2105
2106
2107
2108
2109
2110
2111
2112
2113
2114
2115
2116
2117
2118
2119
2120
2121
2122
2123
2124
2125
2126
2127
2128
2129
2130
2131
2132
2133
2134
2135
2136
2137
2138
2139
2140
2141
2142
2143
2144
2145
2146
2147
2148
2149
2150
2151
2152
2153
2154
2155
2156
2157
2158
2159
2160
2161
2162
2163
2164
2165
2166
2167
2168
2169
2170
2171
2172
2173
2174
2175
2176
2177
2178
2179
2180
2181
2182
2183
2184
2185
2186
2187
2188
2189
2190
2191
2192
2193
2194
2195
2196
21

9 each of the n correlators is arranged to control a
10 corresponding set of equalizer coefficients.

1 63. The equalizer of claim 34 wherein $n > 4$.

2 64. The equalizer of claim 34 further comprising
3 a spectral transformation applied to each data block
4 upstream of the $n - 1$ data shifters and the n finite
5 filters, wherein the spectral transformation is longer than
6 a block of data.

7 65. An equalizer for processing blocks of data
8 comprising:

9 a first data shifter, wherein the first data
10 shifter is arranged to shift the data left by two;

a second data shifter, wherein the second data
shifter is arranged to shift the data left by one;

a third data shifter, wherein the third data
shifter is arranged to shift the data right by one;

a fourth data shifter, wherein the fourth data
shifter is arranged to shift the data right by two;

11 a first finite filter, wherein the first finite
12 filter applies a first set of finite filter coefficients to
13 each of the blocks of data which have been shifted by the
14 first data shifter, wherein ghosts of the blocks of data are
15 not eliminated as a result of the application of the first
16 set of finite filter coefficients;

17 a second finite filter, wherein the second finite
18 filter applies a second set of finite filter coefficients to
19 each of the blocks of data which have been shifted by the
20 second data shifter, wherein ghosts of the blocks of data
21 are not eliminated as a result of the application of the
22 second set of finite filter coefficients;

23 a third finite filter, wherein the third finite
24 filter applies a third set of finite filter coefficients to
25 each of the blocks of data, wherein ghosts of the blocks of
26 data are not eliminated as a result of the application of
27 the third set of finite filter coefficients;

28 a fourth finite filter, wherein the fourth finite
29 filter applies a fourth set of finite filter coefficients to
30 each of the blocks of data which have been shifted by the
31 third data shifter, wherein ghosts of the blocks of data are

32 not eliminated as a result of the application of the fourth
33 set of finite filter coefficients;

34 a fifth finite filter, wherein the fifth finite
35 filter applies a fifth set of finite filter coefficients to
36 each of the blocks of data which have been shifted by the
37 fourth data shifter, wherein ghosts of the blocks of data
38 are not eliminated as a result of the application of the
39 fifth set of finite filter coefficients;

40 an adder arranged to add outputs from the first,
41 second, third, fourth, and fifth finite filters; and,

42 a controller arranged to control the first,
43 second, third, fourth, and fifth sets of finite filter
44 coefficients so that the addition performed by the adder
45 substantially eliminates the ghosts.

1 66. The equalizer of claim 65 wherein the
2 controller comprises a channel estimator that estimates the
3 channel through which the blocks of data are transmitted.

1 67. The equalizer of claim 65 wherein the
2 controller controls the first, second, third, fourth, and
3 fifth sets of finite filter coefficients based upon an
4 output of the adder.

1 68. The equalizer of claim 65 wherein the
2 controller comprises a comparator that performs a comparison
3 based upon an output of the adder and the blocks of data.

1 69. The equalizer of claim 65 wherein the
2 controller comprises:
3

4 a comparator arranged to compare an output of the
5 adder to a reference to form an error;
6

7 a conjugator arranged to conjugate the blocks of
8 data;
9

10 a first data shifter arranged to shift the
11 conjugated blocks of data left by two;
12

1 a second data shifter arranged to shift the
2 conjugated blocks of data left by one;
3

4 a third data shifter arranged to shift the
5 conjugated blocks of data right by one;
6

13 a fourth data shifter arranged to shift the
14 conjugated blocks of data right by two;

15 a first correlator arranged to perform a
16 correlation of the error and the conjugated blocks of data
17 shifted by the first data shifter, wherein the first
18 correlator is arranged to control the first set of finite
19 filter coefficients;

20 a second correlator arranged to perform a
21 correlation of the error and the conjugated blocks of data
22 shifted by the second data shifter, wherein the second
23 correlator is arranged to control the second set of finite
24 filter coefficients;

25 a third correlator arranged to perform a
26 correlation of the error and the conjugated blocks of data,
27 wherein the third correlator is arranged to control the
28 third set of finite filter coefficients;

29 a fourth correlator arranged to perform a
30 correlation of the error and the conjugated blocks of data
31 shifted by the third data shifter, wherein the fourth
32 correlator is arranged to control the fourth set of finite
33 filter coefficients; and,

34 a fifth correlator arranged to perform a
35 correlation of the error and the conjugated blocks of data
36 shifted by the fourth data shifter, wherein the fifth
37 correlator is arranged to control the fifth set of finite
38 filter coefficients.

1 70. The equalizer of claim 69 wherein the
2 controller further comprises a down sampler upstream of each
3 of the first, second, third, fourth, and fifth correlators,
4 wherein the down samplers are arranged to down sample the
5 conjugated blocks of data.

6 71. The equalizer of claim 69 wherein the
7 reference is a training signal.

8 72. The equalizer of claim 69 wherein the
9 reference is sliced data.

10 73. The equalizer of claim 65 wherein the
11 controller comprises:

12 a comparator arranged to compare an output of the
13 adder to a reference to form an error;
14

5 a first data shifter arranged to perform a shift
6 left by two based upon the blocks of data;

7 a second data shifter arranged to perform a shift
8 left by one based upon the blocks of data;

9 a third data shifter arranged to perform a shift
10 right by one based upon the blocks of data;

11 a fourth data shifter arranged to perform a shift
12 right by two based upon the blocks of data;

13 a first correlator arranged to perform a
14 correlation of an output of the first data shifter and the
15 error, wherein the first correlator is arranged to control
16 the first set of finite filter coefficients;

17 a second correlator arranged to perform a
18 correlation of an output of the second data shifter and the
19 error, wherein the second correlator is arranged to control
20 the second set of finite filter coefficients;

21 a third correlator arranged to perform a
22 correlation based upon the blocks of data and the error,
23 wherein the third correlator is arranged to control the
24 third set of finite filter coefficients;

25 a fourth correlator arranged to perform a
26 correlation of an output of the third data shifter and the

27 error, wherein the fourth correlator is arranged to control
28 the fourth set of finite filter coefficients; and,

29 a fifth correlator arranged to perform a
30 correlation of an output of the fourth data shifter and the
31 error, wherein the fifth correlator is arranged to control
32 the fifth set of finite filter coefficients.

1 74. The equalizer of claim 73 wherein the
2 reference is a training signal.

3 75. The equalizer of claim 73 wherein the
4 reference is sliced data.

5 76. The equalizer of claim 65 wherein the
6 controller comprises:

7 a conjugator arranged to conjugate the blocks of
8 data;

a first data shifter arranged to shift the
conjugated blocks of data left by two;

a second data shifter arranged to shift the
conjugated blocks of data left by one;

9 a third data shifter arranged to shift the
10 conjugated blocks of data right by one;

11 a fourth data shifter arranged to shift the
12 conjugated blocks of data right by two;

13 a first correlator arranged to perform a
14 correlation based upon an output of the adder and the
15 conjugated blocks of data shifted by the first data shifter,
16 wherein the first correlator is arranged to control the
17 first set of finite filter coefficients;

18 a second correlator arranged to perform a
19 correlation based upon the output of the adder and the
20 conjugated blocks of data shifted by the second data
21 shifter, wherein the second correlator is arranged to
22 control the second set of finite filter coefficients;

23 a third correlator arranged to perform a
24 correlation based upon the output of the adder and the
25 conjugated blocks of data, wherein the third correlator is
26 arranged to control the third set of finite filter
27 coefficients;

28 a fourth correlator arranged to perform a
29 correlation based upon the output of the adder and the
30 conjugated blocks of data shifted by the third data shifter,

31 wherein the fourth correlator is arranged to control the
32 fourth set of finite filter coefficients; and,

33 a fifth correlator arranged to perform a
34 correlation based upon the output of the adder and the
35 conjugated blocks of data shifted by the fourth data
36 shifter, wherein the fifth correlator is arranged to control
37 the fifth set of finite filter coefficients.

77. The equalizer of claim 65 wherein the
controller comprises:

a first data shifter arranged to perform a shift
left by two operation based upon the blocks of data;

a second data shifter arranged to perform a shift
left by one operation based upon the blocks of data;

a third data shifter arranged to perform a shift
right by one operation based upon the blocks of data;

a fourth data shifter arranged to perform a shift
right by two operation based upon the blocks of data;

a first correlator arranged to perform a
correlation based upon an output of the adder and an output
of the first data shifter, wherein the first correlator is

14 arranged to control the first set of finite filter
15 coefficients;

16 a second correlator arranged to perform a
17 correlation based upon the output of the adder and an output
18 of the second data shifter, wherein the second correlator is
19 arranged to control the second set of finite filter
20 coefficients;

21 a third correlator arranged to perform a
22 correlation based upon the output of the adder and the
23 blocks of data, wherein the third correlator is arranged to
24 control the third set of finite filter coefficients;

25 a fourth correlator arranged to perform a
26 correlation based upon the output of the adder and an output
27 of the third data shifter, wherein the fourth correlator is
28 arranged to control the fourth set of finite filter
29 coefficients; and,

30 a fifth correlator arranged to perform a
31 correlation based upon the output of the adder and an output
32 of the fourth data shifter, wherein the fifth correlator is
33 arranged to control the fifth set of finite filter
34 coefficients.

1 78. The equalizer of claim 65 further comprising
2 a pre-processor that applies pre-processor coefficients to
3 each block of data upstream of the first, second, third, and
4 fourth data shifters and upstream of the third finite
5 filter, wherein the controller controls a width of the pre-
6 processing coefficients so that the width is substantially
7 coincident with the width of a block of data and an interval
8 between a block of data and a ghost.

9 79. The equalizer of claim 78 wherein the
10 controller comprises a channel estimator that estimates the
11 channel through which the blocks of data are transmitted.

12 80. The equalizer of claim 78 wherein the
13 controller controls the first, second, third, fourth, and
14 fifth sets of finite filter coefficients based upon an
15 output of the adder.

1 81. The equalizer of claim 78 wherein the
2 controller comprises a comparator that performs a comparison
3 based upon an output of the adder and the blocks of data.

1 82. The equalizer of claim 78 wherein the
2 controller comprises:

3 a comparator arranged to compare an output of the
4 adder to a reference to form an error;

5 a conjugator arranged to conjugate the blocks of
6 data;

7 a first data shifter arranged to shift the
8 conjugated blocks of data left by two;

9 a second data shifter arranged to shift the
10 conjugated blocks of data left by one;

11 a third data shifter arranged to shift the
12 conjugated blocks of data right by one;

13 a fourth data shifter arranged to shift the
14 conjugated blocks of data right by two;

15 a first correlator arranged to perform a
16 correlation of the error and the conjugated blocks of data
17 shifted by the first data shifter, wherein the first
18 correlator is arranged to control the first set of finite
19 filter coefficients;

20 a second correlator arranged to perform a
21 correlation of the error and the conjugated blocks of data
22 shifted by the second data shifter, wherein the second

23 correlator is arranged to control the second set of finite
24 filter coefficients;

25 a third correlator arranged to perform a
26 correlation of the error and the conjugated blocks of data,
27 wherein the third correlator is arranged to control the
28 third set of finite filter coefficients;

29 a fourth correlator arranged to perform a
30 correlation of the error and the conjugated blocks of data
31 shifted by the third data shifter, wherein the fourth
32 correlator is arranged to control the fourth set of finite
33 filter coefficients; and,

34 a fifth correlator arranged to perform a
35 correlation of the error and the conjugated blocks of data
36 shifted by the fourth data shifter, wherein the fifth
37 correlator is arranged to control the fifth set of finite
38 filter coefficients.

1 83. The equalizer of claim 82 wherein the
2 controller further comprises a down sampler upstream of each
3 of the first, second, third, fourth, and fifth correlators,
4 wherein the down samplers are arranged to down sample the
5 conjugated blocks of data prior.

1 84. The equalizer of claim 82 wherein the
2 reference is a training signal.

1 85. The equalizer of claim 82 wherein the
2 reference is sliced data.

1 86. The equalizer of claim 78 wherein the
2 controller comprises:

3 a comparator arranged to compare an output of the
4 adder to a reference to form an error;

5 a first data shifter arranged to perform a shift
6 left by two based upon the blocks of data;

7 a second data shifter arranged to perform a shift
8 left by one based upon the blocks of data;

9 a third data shifter arranged to perform a shift
10 right by one based upon the blocks of data;

11 a fourth data shifter arranged to perform a shift
12 right by two based upon the blocks of data;

13 a first correlator arranged to perform a
14 correlation of the error and an output of the first data

15 shifter, wherein the first correlator is arranged to control
16 the first set of finite filter coefficients;

17 a second correlator arranged to perform a
18 correlation of the error and an output of the second data
19 shifter, wherein the second correlator is arranged to
20 control the second set of finite filter coefficients;

21 a third correlator arranged to perform a
22 correlation based upon the error and the blocks of data,
23 wherein the third correlator is arranged to control the
24 third set of finite filter coefficients;

25 a fourth correlator arranged to perform a
26 correlation of the error and an output of the third data
27 shifter, wherein the fourth correlator is arranged to
28 control the fourth set of finite filter coefficients; and,

29 a fifth correlator arranged to perform a
30 correlation of the error and an output of the fourth data
31 shifter, wherein the fifth correlator is arranged to control
32 the fifth set of finite filter coefficients.

1 87. The equalizer of claim 86 wherein the
2 reference is a training signal.

1 88. The equalizer of claim 86 wherein the
2 reference is sliced data.

1 89. The equalizer of claim 78 wherein the
2 controller comprises:

3 a conjugator arranged to conjugate the blocks of
4 data;

5 a first data shifter arranged to shift the
6 conjugated blocks of data left by two;

7 a second data shifter arranged to shift the
8 conjugated blocks of data left by one;

9 a third data shifter arranged to shift the
10 conjugated blocks of data right by one;

11 a fourth data shifter arranged to shift the
12 conjugated blocks of data right by two;

13 a first correlator arranged to perform a
14 correlation based upon an output of the adder and the
15 conjugated blocks of data shifted by the first data shifter,
16 wherein the first correlator is arranged to control the
17 first set of finite filter coefficients;

18 a second correlator arranged to perform a
19 correlation based upon the output of the adder and the

20 conjugated blocks of data shifted by the second data
21 shifter, wherein the second correlator is arranged to
22 control the second set of finite filter coefficients;

23 a third correlator arranged to perform a
24 correlation based upon the output of the adder and the
25 conjugated blocks of data, wherein the third correlator is
26 arranged to control the third set of finite filter
27 coefficients;

28 a fourth correlator arranged to perform a
29 correlation based upon the output of the adder and the
30 conjugated blocks of data shifted by the third data shifter,
31 wherein the fourth correlator is arranged to control the
32 fourth set of finite filter coefficients; and,

33 a fifth correlator arranged to perform a
34 correlation based upon the output of the adder and the
35 conjugated blocks of data shifted by the fourth data
36 shifter, wherein the fifth correlator is arranged to control
37 the fifth set of finite filter coefficients.

1 90. The equalizer of claim 78 wherein the
2 controller comprises:

3 a first data shifter arranged to perform a shift
4 left by two operation based upon the blocks of data;

5 a second data shifter arranged to perform a shift
6 left by one operation based upon the blocks of data;

7 a third data shifter arranged to perform a shift
8 right by one operation based upon the blocks of data;

9 a fourth data shifter arranged to perform a shift
10 right by two operation based upon the blocks of data;

11 a first correlator arranged to perform a
12 correlation based upon an output of the adder and an output
13 of the first data shifter, wherein the first correlator is
14 arranged to control the first set of finite filter
15 coefficients;

16 a second correlator arranged to perform a
17 correlation based upon the output of the adder and an output
18 of the second data shifter, wherein the second correlator is
19 arranged to control the second set of finite filter
20 coefficients;

21 a third correlator arranged to perform a
22 correlation based upon the output of the adder and the

23 blocks of data, wherein the third correlator is arranged to
24 control the third set of finite filter coefficients;

25 a fourth correlator arranged to perform a
26 correlation based upon the output of the adder and an output
27 of the third data shifter, wherein the fourth correlator is
28 arranged to control the fourth set of finite filter
29 coefficients; and,

30 a fifth correlator arranged to perform a
31 correlation based upon the output of the adder and an output
32 of the fourth data shifter, wherein the fifth correlator is
33 arranged to control the fifth set of finite filter
34 coefficients.

91. The equalizer of claim 78 wherein the pre-
processing coefficients are curved.

1 92. The equalizer of claim 78 wherein the pre-
2 processing coefficients are curved substantially according
3 to a function $1/(2 - \cos(t))$.

4 93. The equalizer of claim 65 further comprising
5 a spectral transformation applied to each data block
6 upstream of the first, second, third, and fourth data
7 shifters and the first, second, third, fourth, and fifth
8 finite filter, wherein the spectral transformation is longer
9 than a block of data.

ABSTRACT OF THE DISCLOSURE

An equalizer for processing blocks of data includes $n - 1$ data shifters, n finite filters, an adder, and a controller. Each of the $n - 1$ data shifters shifts the blocks of data. One of the n finite filters applies a corresponding set of finite filter coefficients to the blocks of data, and each of the other $n - 1$ finite filters applies a set of finite filter coefficients to a corresponding output of the $n - 1$ data shifters. Ghosts of the blocks of data are not eliminated as a result of the application of the n sets of finite filter coefficients corresponding to the n finite filters, and $n > 2$. The adder is arranged to add outputs from the n finite filters. The controller is arranged to control the sets of finite filter coefficients corresponding to the n finite filters so that the addition performed by the adder substantially eliminates the ghosts.

7163
7164
7147
7165

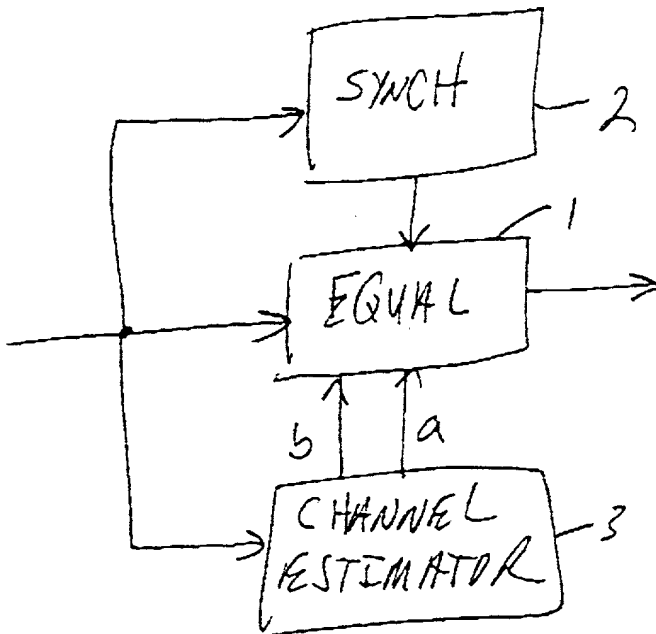


FIGURE 1

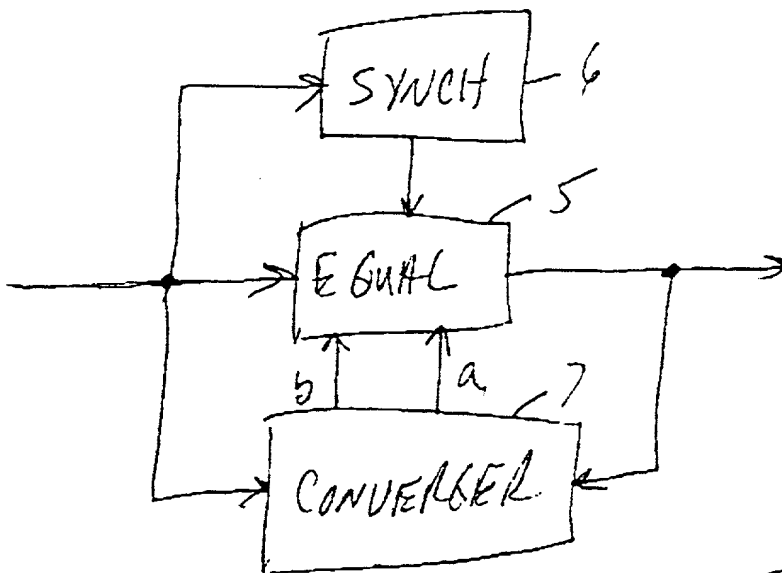


FIGURE 2

0054597.040700

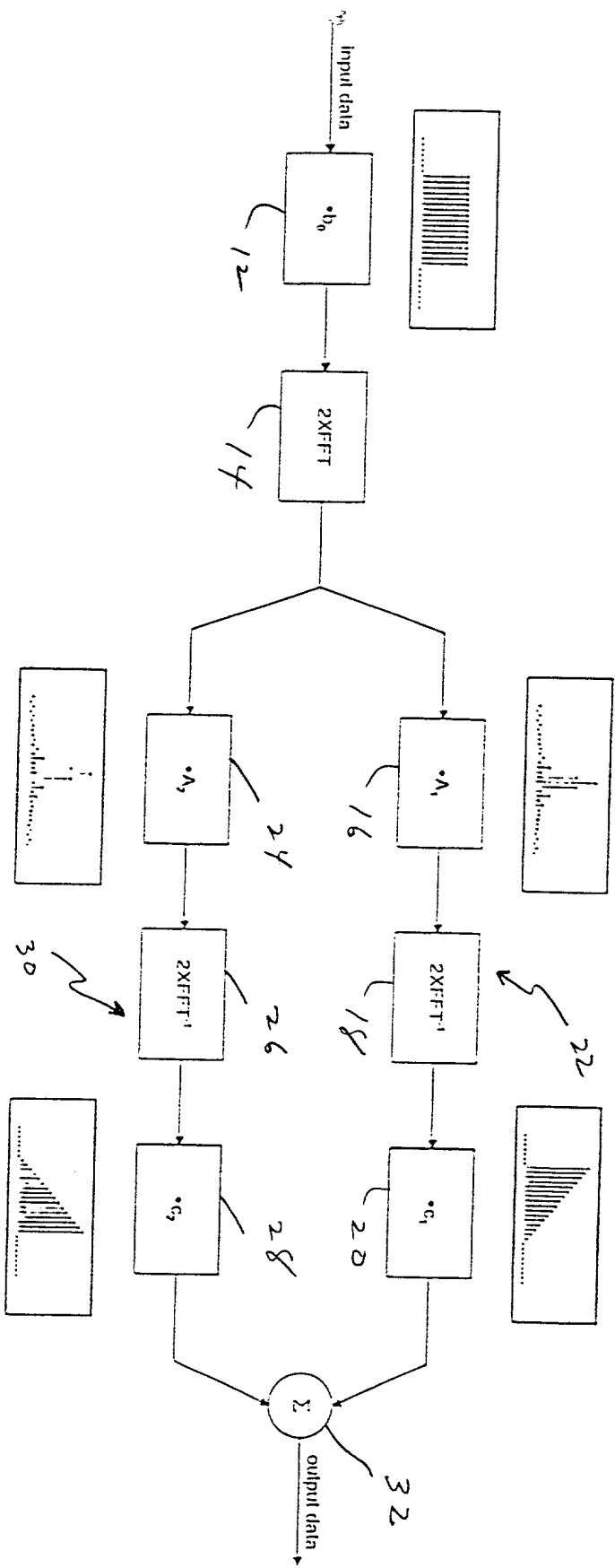


Fig. 3

D=1/16
G=1
10.97 dB



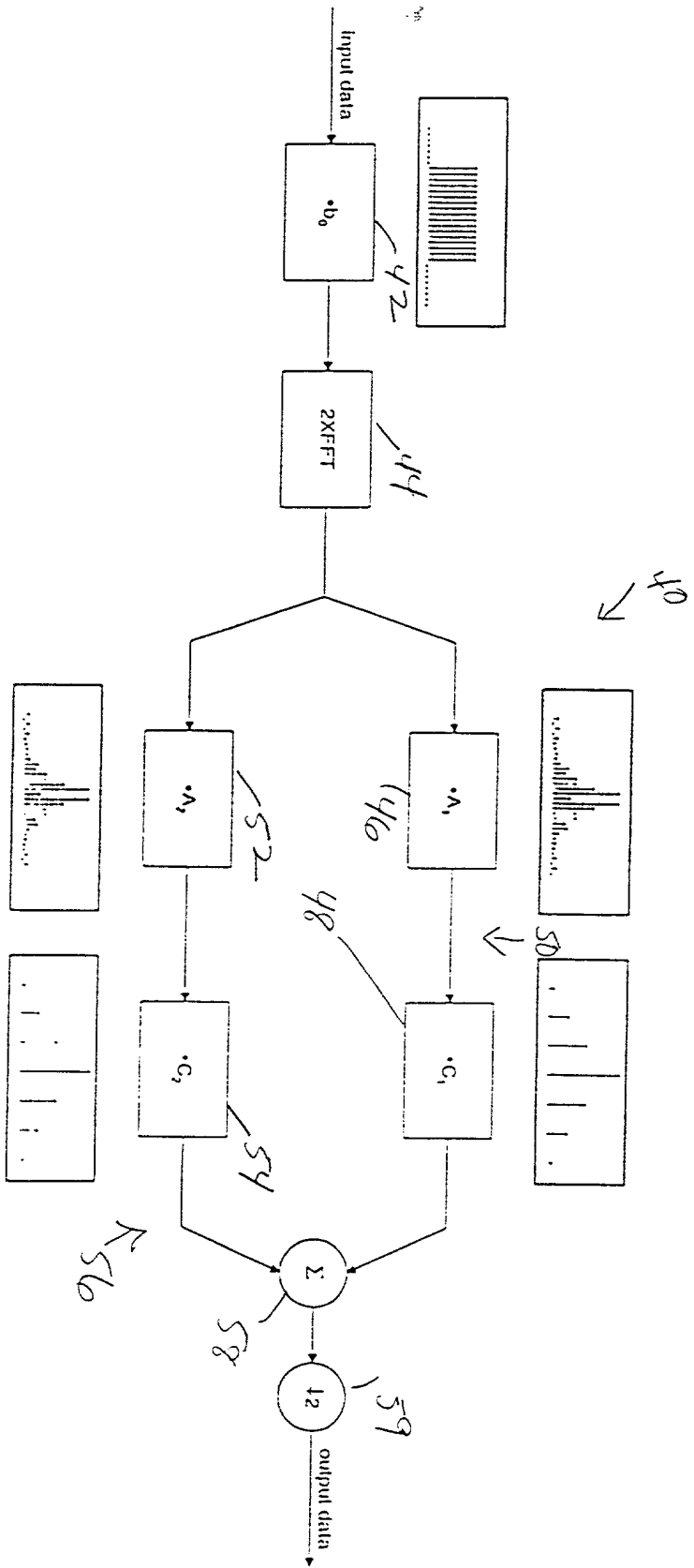


Fig. 5

DELAY=1/16, GAIN=1, TARGET THRESHOLD=15 DB

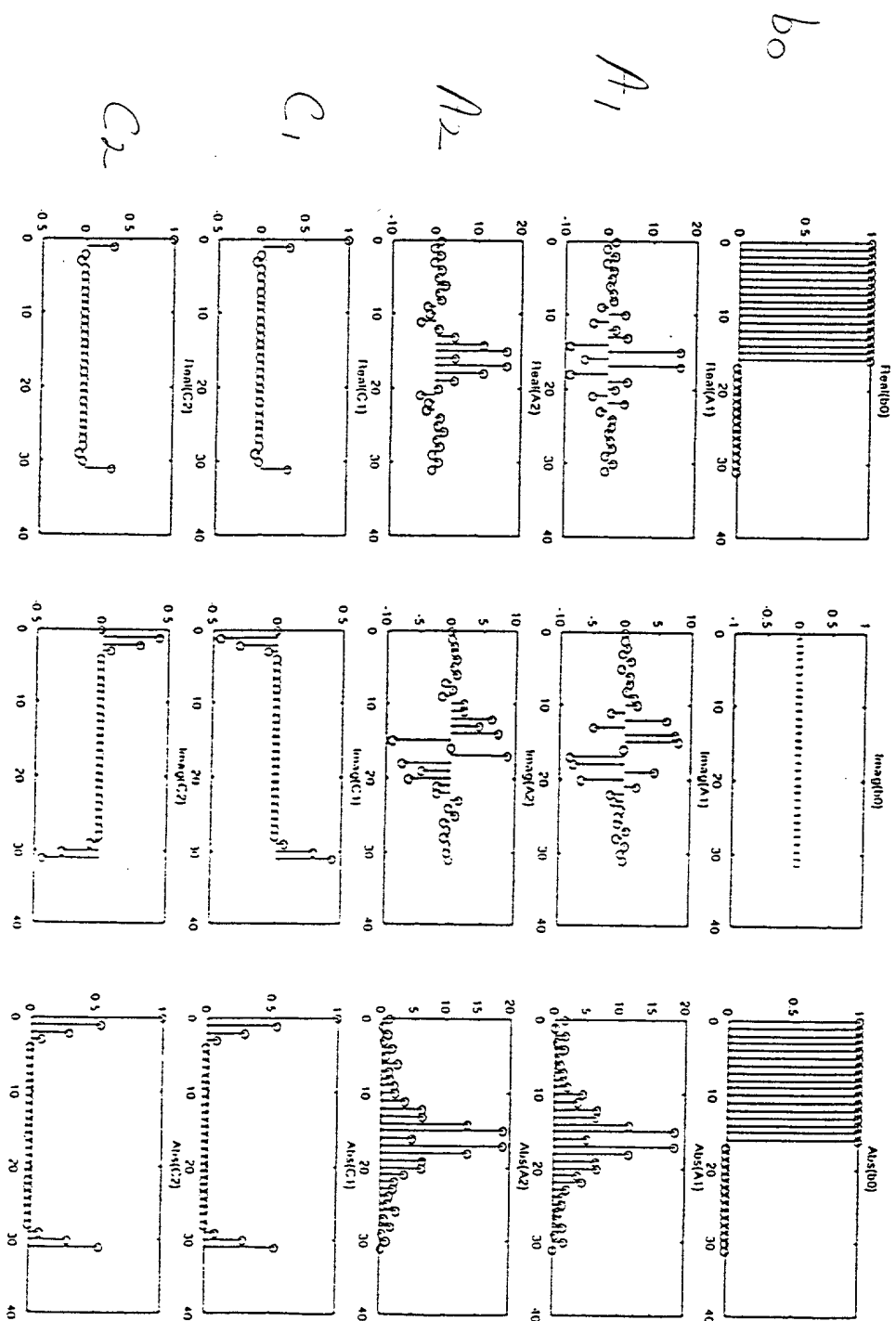


Fig. 6

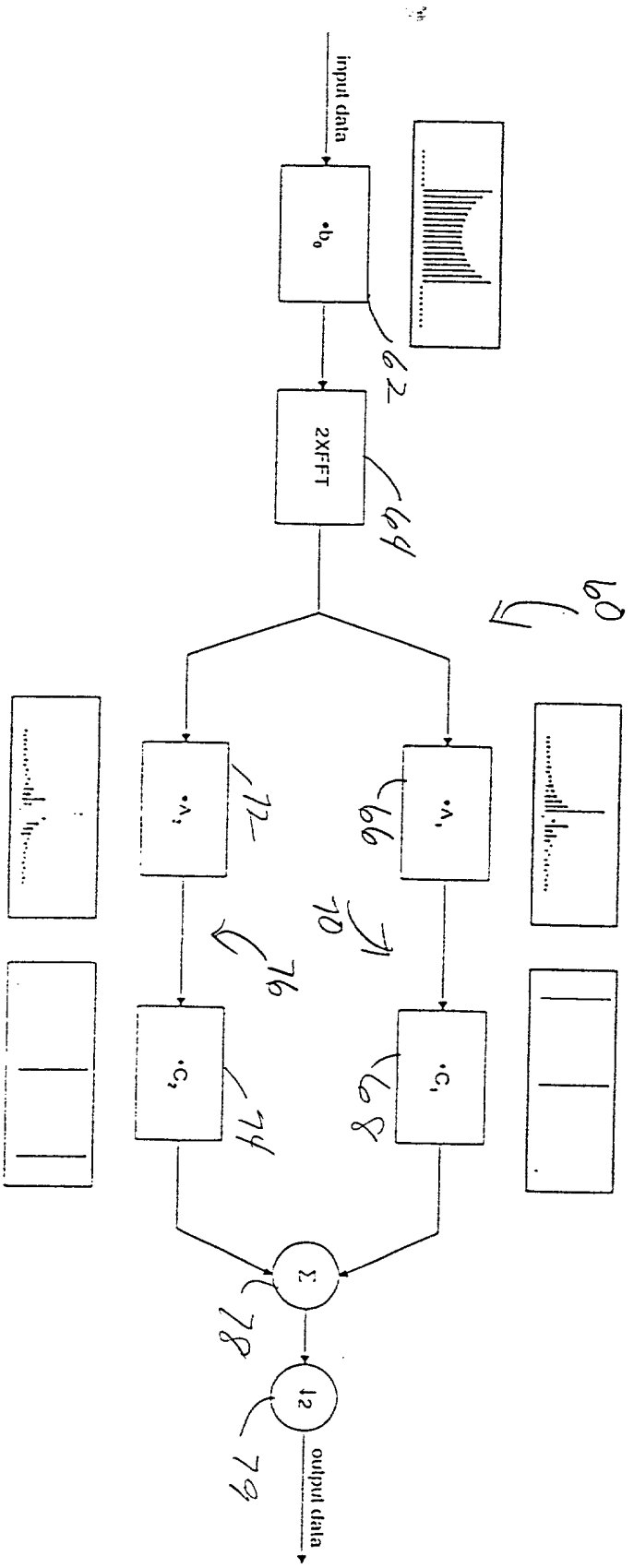


Fig. 7

09545597 040700

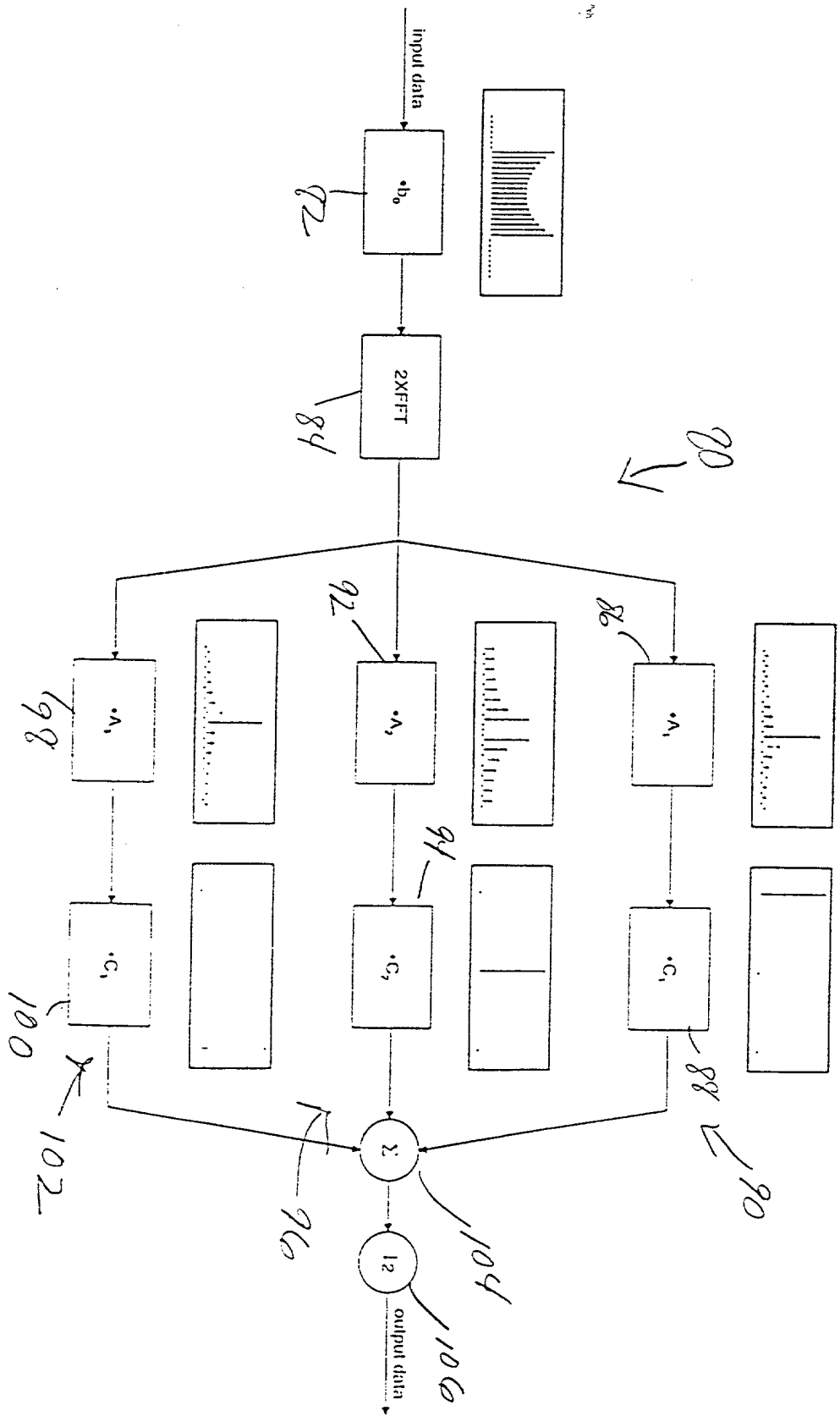


Fig. 8

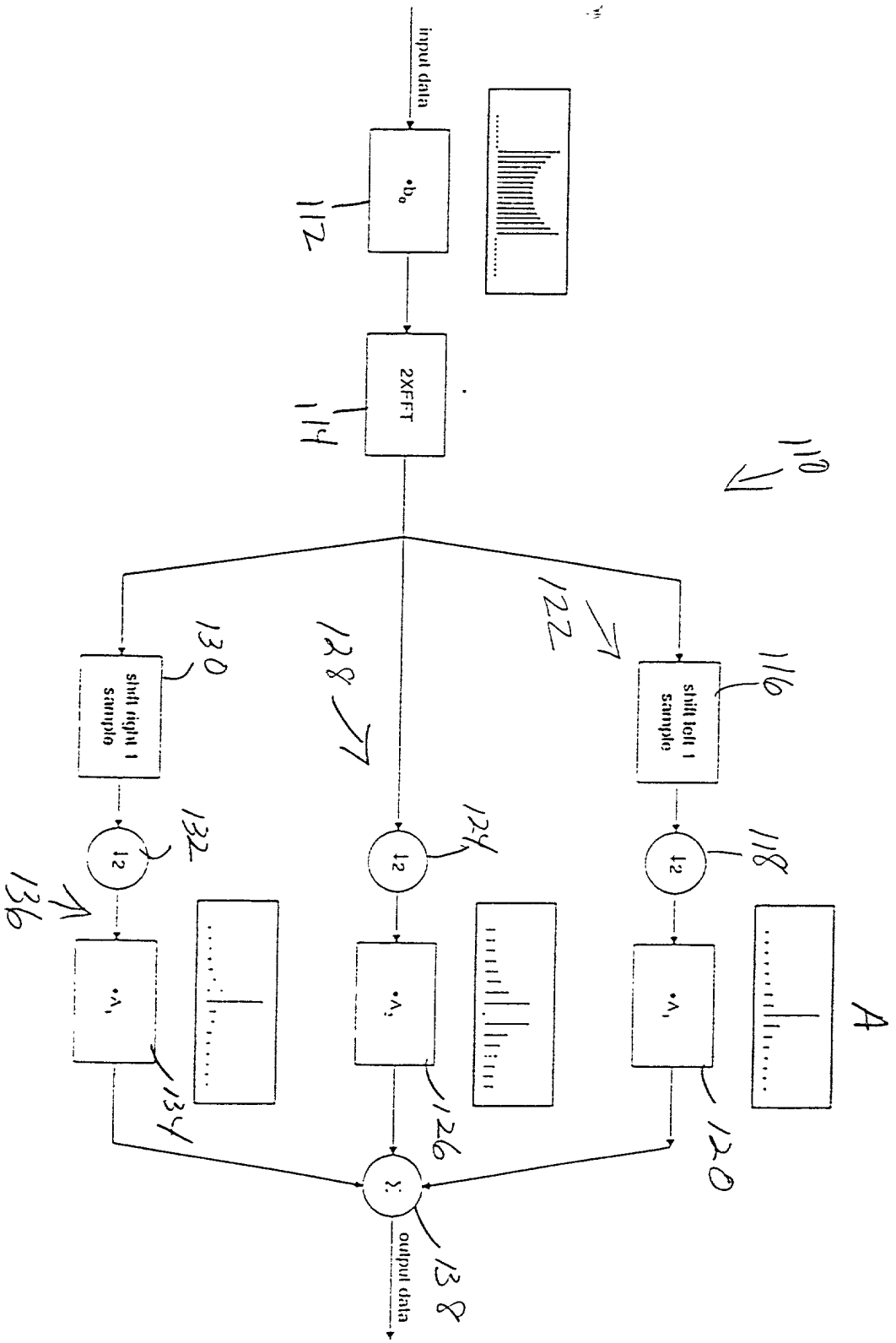
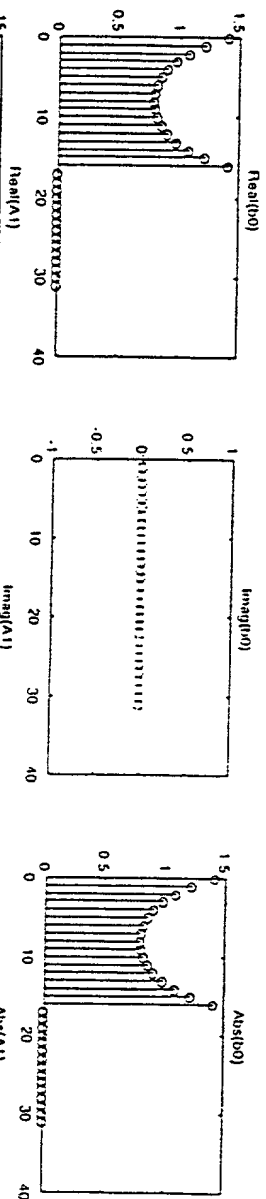


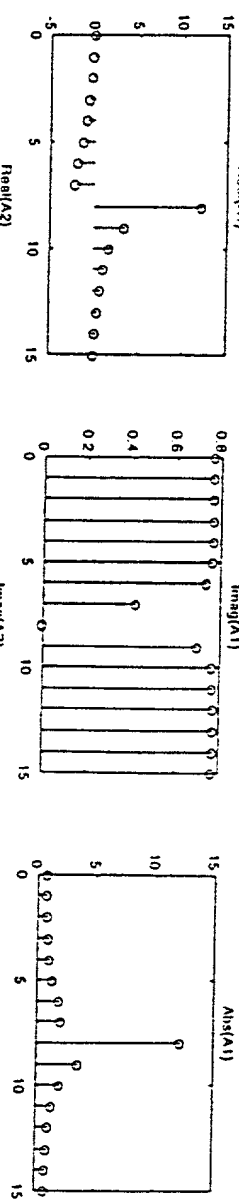
Fig. 9

DELAY=1/16, GAIN=1, TARGET THRESHOLD=15 DB

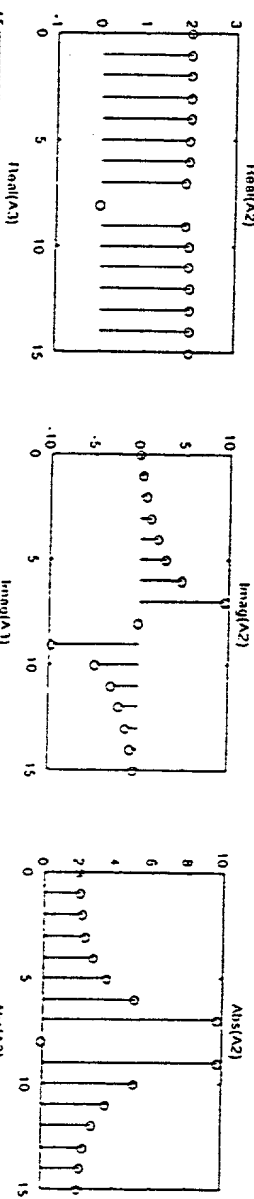
b0
D=1/16
G=1
10.91 dB



A1



A2



A3

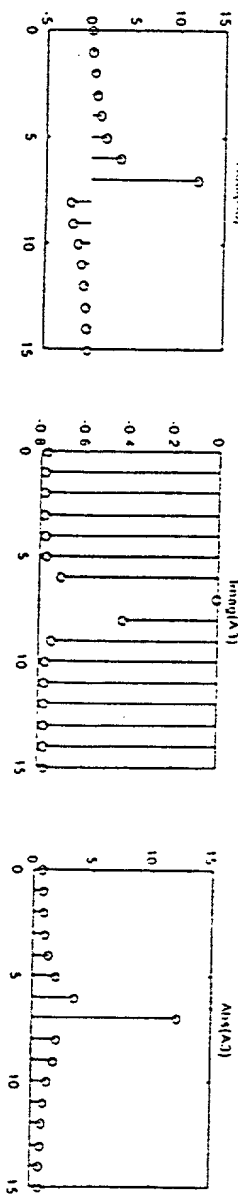


Fig. 11

054455Z 040700



144

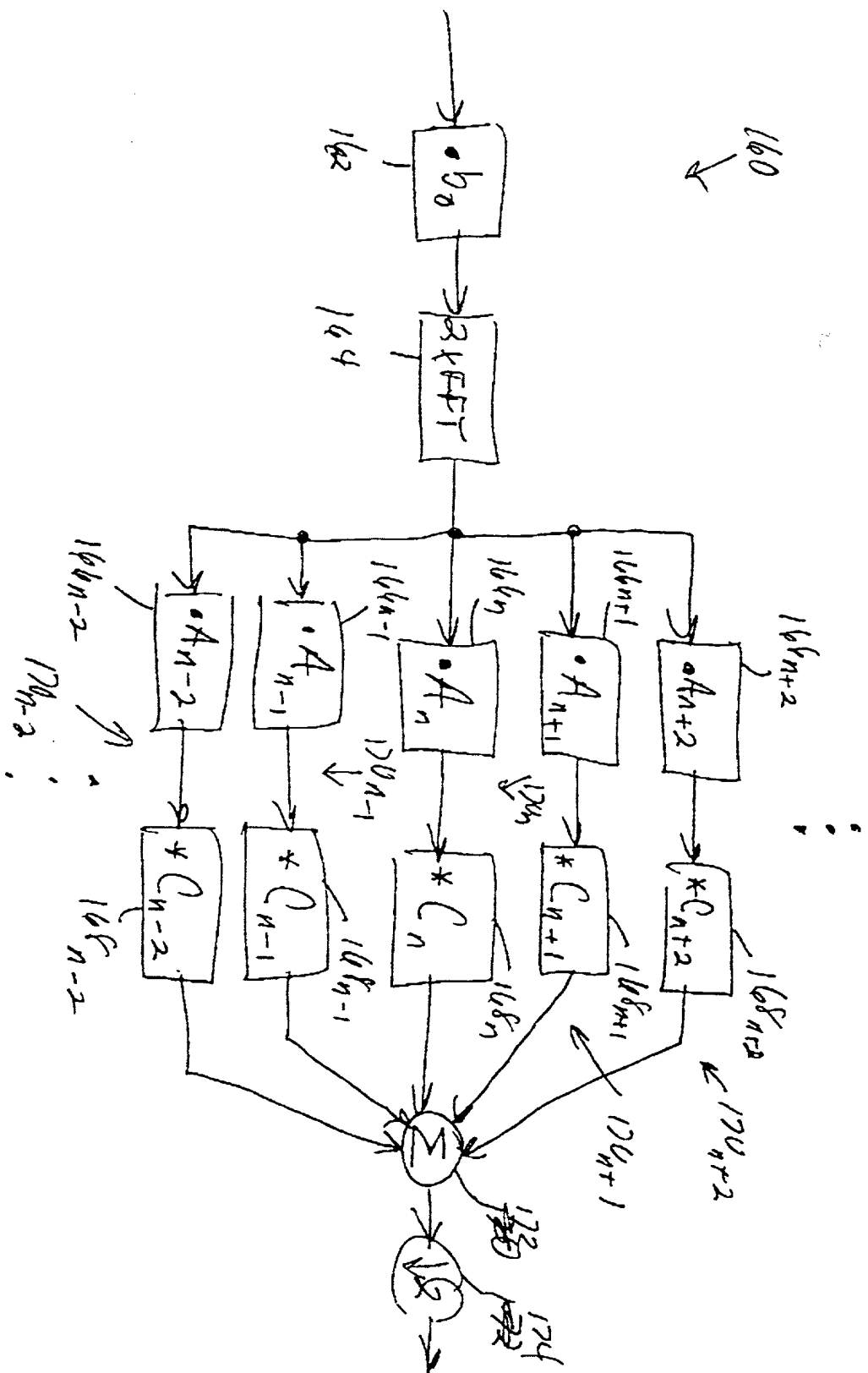
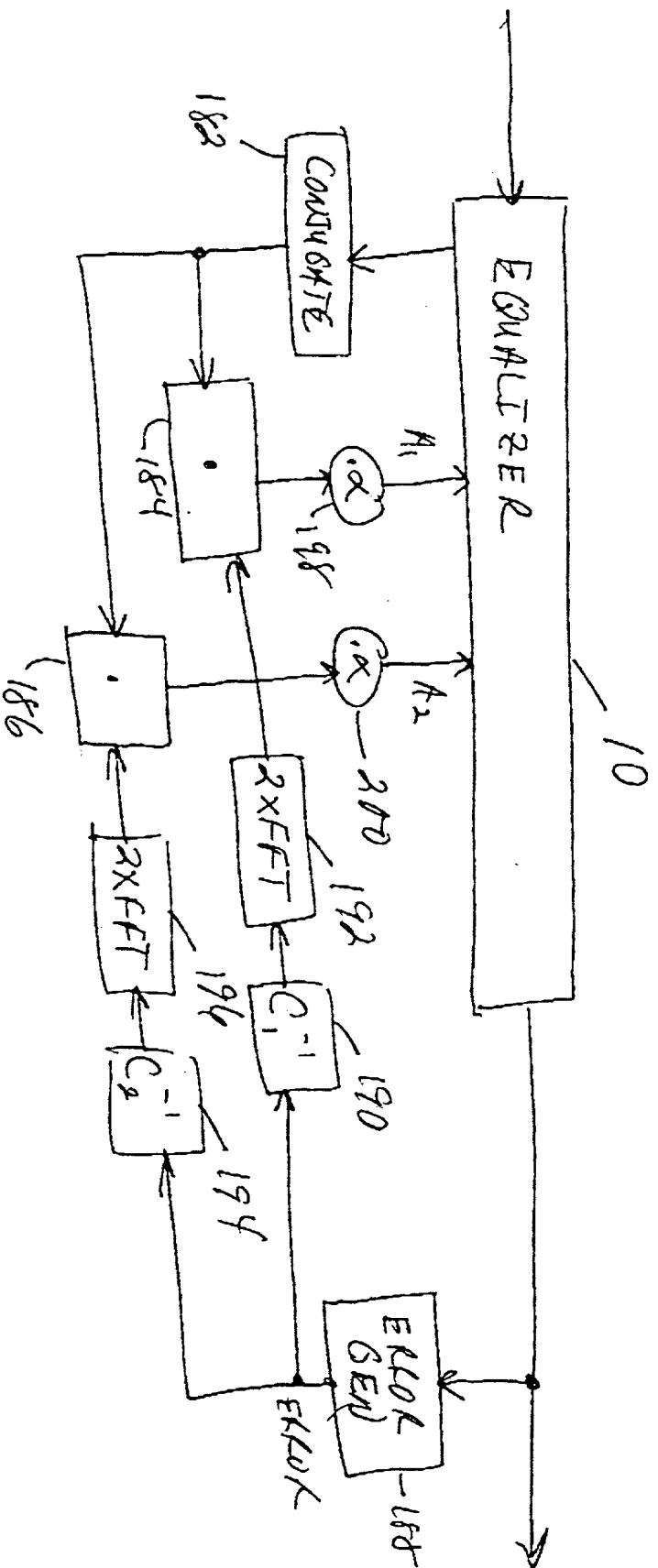


TABLE 13



180 →

FIGURE 14

09545597.040700

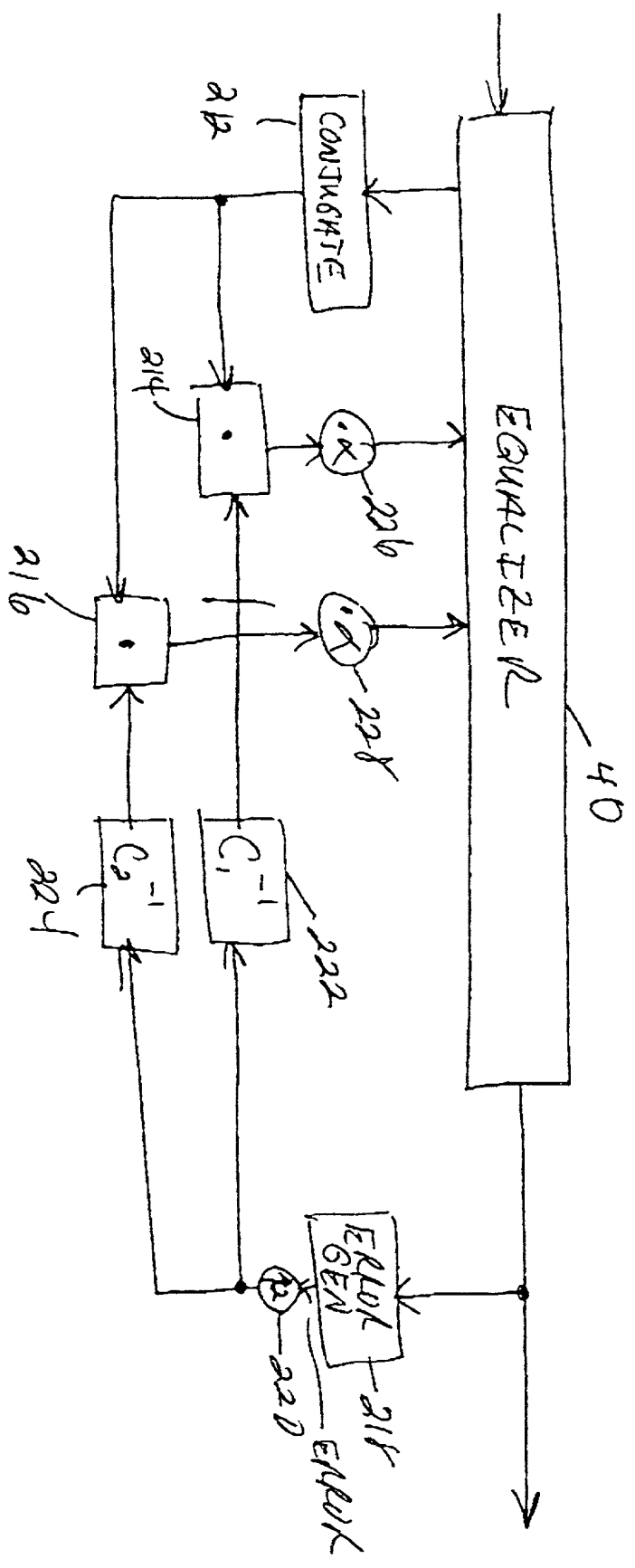
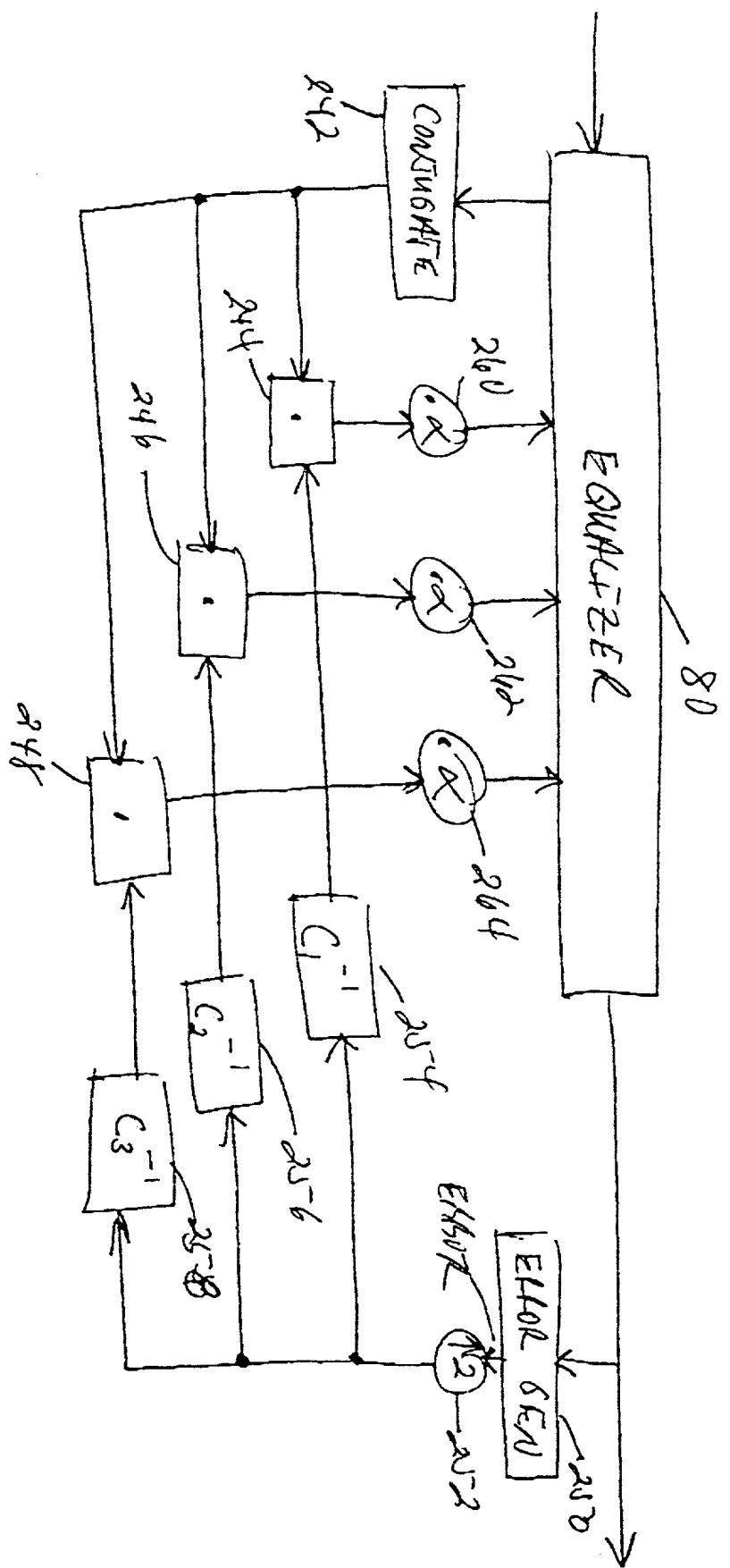


FIGURE 15

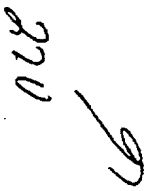
09545597.040700



240 →

FIGURE 16

09545597.040700



FT6442

Abstract

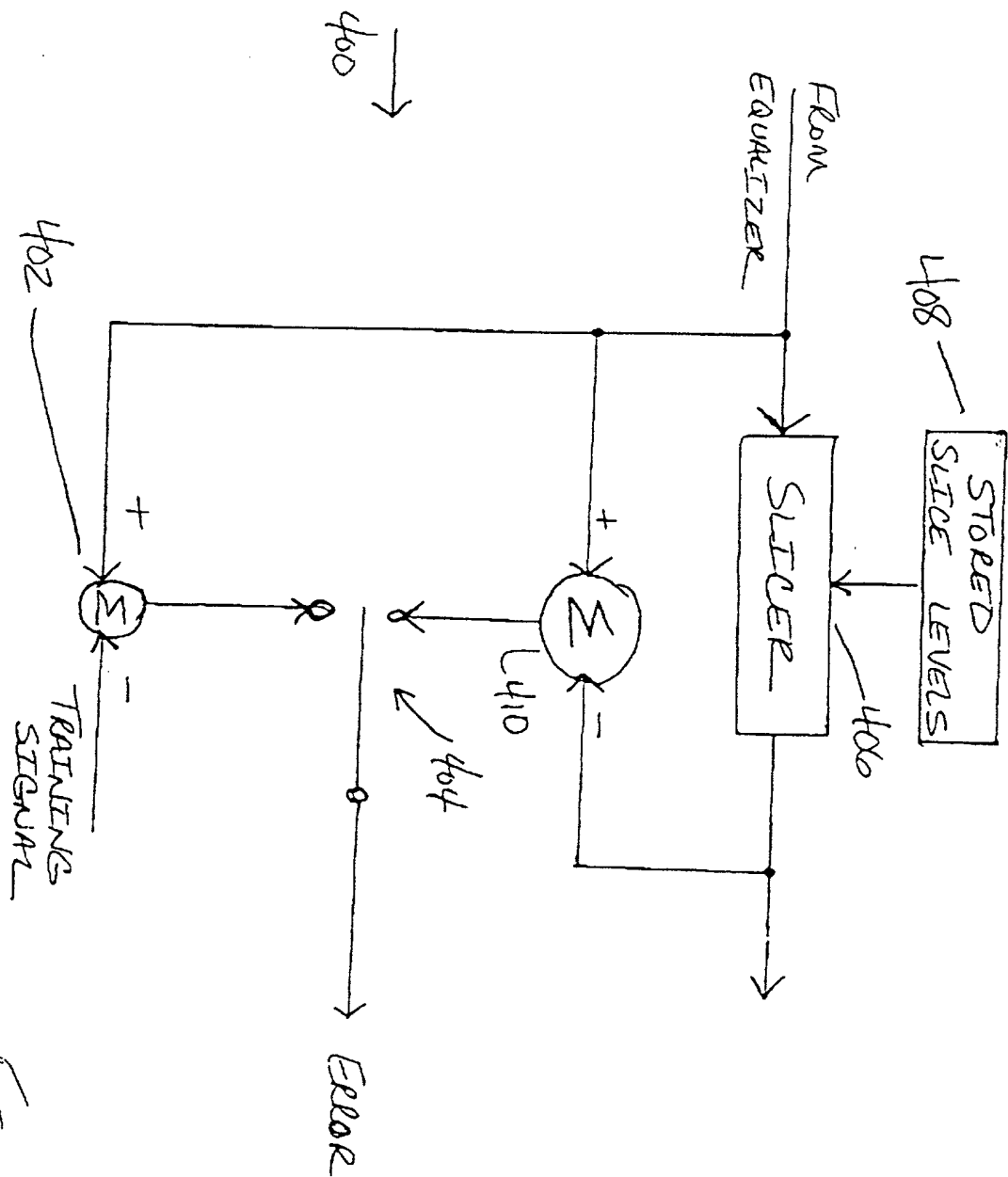


FIGURE
18

09545597 .040700

DECLARATION FOR PATENT APPLICATION AND POWER OF ATTORNEY

As a below named inventor, I hereby declare that my residence, post office address and citizenship are as stated below next to my name; I believe that I am the original, first and sole inventor (if only one name is listed below) or an original, first and joint inventor (if plural names are listed below) of the subject matter which is claimed and for which a patent is sought on the invention entitled **"Multipath Ghost Eliminating Equalizer With Optimum Noise Enhancement,"** the specification of which (check one): ☒ is attached hereto; ☐ was filed on _____ as Application Serial No. _____ and was amended on _____ (if applicable) _____ was filed as PCT International Application No. _____ on _____ and was amended under Article 19 on _____ (if applicable). I hereby state that I have reviewed and understand the contents of the above-identified specification, including the claims, as amended by any amendment(s) referred to above. I acknowledge the duty to disclose to the Patent and Trademark Office all information known to me to be material to patentability as defined in 37 C.F.R. § 1.56.

I hereby claim foreign priority benefits under 35 U.S.C. §119 of any foreign application(s) for patent or inventor's certificate or of any PCT international application(s) designating at least one country other than the United States of America listed below and have also identified below any foreign application(s) for patent or inventor's certificate or any PCT international application(s) designating at least one country other than the United States of America filed by me on the same subject matter having a filing date before that of the application(s) of which priority is claimed:

Priority Claimed

(Application Serial Number) (Country) (Day/Month/Year Filed)

☐ Yes ☐ No

(Application Serial Number) (Country) (Day/Month/Year Filed)

☐ Yes ☐ No

I hereby claim the benefit under 35 U.S.C. §119(e) of any United States provisional application(s) listed below:

(Application Serial Number) (Day/Month/Year Filed)

(Application Serial Number) (Day/Month/Year Filed)

I hereby claim the benefit under 35 U.S.C. §120 of any United States application(s) or PCT international application(s) designating the United States of America listed below and, insofar as the subject matter of each of the claims of this application is not disclosed in the prior application(s) in the manner provided by the first paragraph of 35 U.S.C. §112, I acknowledge the duty to disclose to the Office all information known to me to be material to patentability as defined in 37 U.S.C. §1.56 which occurred between the filing date of the prior application(s) and the national or PCT international filing date of this application:

(Application Serial Number) (Day/Month/Year Filed) (Status - Patented, Pending or Abandoned)

(Application Serial Number) (Day/Month/Year Filed) (Status - Patented, Pending or Abandoned)

I hereby declare that all statements made herein of my own knowledge are true and that all statements made on information and belief are believed to be true; and further that these statements were made with the knowledge that willful false statements and the like so made are punishable by fine or imprisonment, or both, under 18 U.S.C. §1001 and that such willful false statements may jeopardize the validity of the application or any patent issued thereon.

POWER OF ATTORNEY: I hereby appoint as my attorneys, with full powers of substitution and revocation, to prosecute this application and transact all business in the Patent and Trademark Office connected therewith:

Jack Kail

Registration No. 27,566

SEND CORRESPONDENCE TO:

FIRM NAME

PHONE NO.

STREET

CITY & STATE

ZIP CODE

Jack Kail
Zenith Electronics Corporation

847-391-8011

1000 Milwaukee Avenue

Glenview, IL

60025

Full Name of First or Sole Inventor Richard W. Citta	Citizenship USA
Residence Address - Street 739 N. Columbian	Post Office Address - Street 739 N. Columbian
City (Zip) Oak Park, 60302	City (Zip) Oak Park, 60302
State or Country Illinois	State or Country Illinois
Date 4-6-00	Signature <i>Richard W. Citta</i>
Full Name of Second Inventor Scott M. LoPresto	Citizenship USA
Residence Address - Street 2361 S. Lexington Dr. - Apt. 206	Post Office Address - Street 2361 S. Lexington Dr. - Apt. 206
City (Zip) Mt. Prospect, 60056	City (Zip) Mt. Prospect, 60056
State or Country Illinois	State or Country Illinois
Date 4/6/2000	Signature <i>Scott M. LoPresto</i>
Full Name of Third Inventor Jingsong Xia	Citizenship China
Residence Address - Street 1044 Evergreen Street	Post Office Address - Street 1044 Evergreen Street
City (Zip) Mundelein 60060	City (Zip) Mundelein 60060
State or Country Illinois	State or Country Illinois
Date 04/06/2000	Signature <i>Jingsong Xia</i>